09755286_CLS.txt

Most Frequently Occurring Classifications of Patents Returned From A Search of 09755286 on June 07, 2004

Original Classifications

5	706/44
4	707/101
4	707/102
3	324/318
3	707/4
2	250/307
2	257/440
2	345/522
2	428/64.2
2	600/420
2	703/2
2	706/25
2	707/1
2	707/3
2	711/100
2	712/20
2	717/107
2	717/121

Cross-Reference Classifications

4	345/835
4	707/104.1
3	257/225
3	706/60
3	707/100
3	707/200
3	717/109
2	250/306
2	257/184
2	257/251
2	257/443
2	257/E27.159
2	345/441
2	345/442
2	345/854
2	369/283
2	369/288
2	370/472
2	428/64.4
2	428/64.5
2	428/64.6
2	428/913
2	430/270.13

2 430/495.1
2 430/945
2 607/122
2 702/28
2 706/44
2 707/1
2 709/203
2 712/11
2 712/209
2 712/21
2 712/22
2 715/500.1
2 715/514
2 715/531

Combined Classifications
7 706/44
5 707/101
5 707/102
4 345/835
4 707/1
4 707/104.1
4 707/200
3 257/225
3 324/318
3 703/2
3 706/60
3 707/100
3 707/3
3 707/4
3 711/100
3 715/531
3 717/109
2 250/306
2 250/307
2 257/184
2 257/251
2 257/440
2 257/443
2 257/E27.159
2 324/309
2 345/441
2 345/442
2 345/522
2 345/853
2 345/854
2 369/283
2 369/288

2 370/472
2 428/64.2
2 428/64.4
2 428/64.5
2 428/64.6
2 428/913
2 430/270.13
2 430/495.1
2 430/945
2 600/420
2 607/122
2 702/28
2 703/5
2 706/25
2 707/10
2 707/2
2 707/201
2 709/203
2 709/236
2 712/11
2 712/20
2 712/209
2 712/21
2 712/22
2 715/500.1
2 715/512
2 715/514
2 716/18
2 717/107
2 717/121

Titles of Most Frequently Occurring Classifications of Patents Returned

From A Search of 09755286 on June 07, 2004

7 706/44 (5 OR, 2 XR)
Class 706 : DATA PROCESSING: ARTIFICIAL INTELLIGENCE
706/15 NEURAL NETWORK
706/44 .Neural simulation environment

5 707/101 (4 OR, 1 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/100 DATABASE SCHEMA OR DATA STRUCTURE
707/101 .Manipulating data structure (e.g., compression, compaction, compilation)

5 707/102 (4 OR, 1 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/100 DATABASE SCHEMA OR DATA STRUCTURE
707/102 .Generating database or data structure (e.g., via user interface)

4 345/835 (0 OR, 4 XR)
Class 345 : COMPUTER GRAPHICS PROCESSING, OPERATOR INTERFACE PROCESSING, AND SELECTIVE VISUAL DISPLAY SYSTEMS
345/700 OPERATOR INTERFACE (E.G., GRAPHICAL USER INTERFACE)
345/764 .On-screen workspace or object
345/810 ..Menu or selectable iconic array (e.g., palette)
345/835 ...Selectable iconic array

4 707/1 (2 OR, 2 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/1 DATABASE OR FILE ACCESSING

4 707/104.1 (0 OR, 4 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/100 DATABASE SCHEMA OR DATA STRUCTURE
707/104.1 .Application of database or data structure (e.g., distributed, multimedia, image)

4 707/200 (1 OR, 3 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/200 FILE OR DATABASE MAINTENANCE

3 257/225 (0 OR, 3 XR)
Class 257 : ACTIVE SOLID-STATE DEVICES
257/213 FIELD EFFECT DEVICE
257/215 .Charge transfer device
257/225 ..Non-electrical input responsive (e.g., light responsive imager, input programmed by size of storage sites for use as a read-only memory, etc.)

3 324/318 (3 OR, 0 XR)
Class 324 : ELECTRICITY: MEASURING AND TESTING
324/300 PARTICLE PRECESSION RESONANCE
324/318 .Spectrometer components

3 703/2 (2 OR, 1 XR)
Class 703 : DATA PROCESSING: STRUCTURAL DESIGN, MODELING, SIMULATION, AND EMULATION
703/2 MODELING BY MATHEMATICAL EXPRESSION

3 706/60 (0 OR, 3 XR)
Class 706 : DATA PROCESSING: ARTIFICIAL INTELLIGENCE
706/45 KNOWLEDGE PROCESSING SYSTEM
706/59 .Creation or modification
706/60 ..Expert system shell or tool

3 707/100 (0 OR, 3 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/100 DATABASE SCHEMA OR DATA STRUCTURE

3 707/3 (2 OR, 1 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/1 DATABASE OR FILE ACCESSING
707/3 .Query processing (i.e., searching)

3 707/4 (3 OR, 0 XR)
Class 707 : DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES
707/1 DATABASE OR FILE ACCESSING
707/3 .Query processing (i.e., searching)
707/4 ..Query formulation, input preparation, or translation

3 711/100 (2 OR, 1 XR)
Class 711 : ELECTRICAL COMPUTERS AND DIGITAL PROCESSING SYSTEMS: MEMORY
711/100 STORAGE ACCESSING AND CONTROL

3 715/531 (1 OR, 2 XR)
Class 715 : DATA PROCESSING: PRESENTATION PROCESSING OF DOCUMENT
715/530 EDIT, COMPOSITION, OR STORAGE CONTROL
715/531 .Text

3 717/109 (0 OR, 3 XR)
Class 717 : DATA PROCESSING: SOFTWARE DEVELOPMENT, INSTALLATION, AND MANAGEMENT
717/100 SOFTWARE PROGRAM DEVELOPMENT TOOL (E.G., INTEGRATED CASE TOOL OR STAND-ALONE DEVEL
OPMENT TOOL)
717/106 .Code generation
717/109 ..Visual

2 250/306 (0 OR, 2 XR)
Class 250 : RADIANT ENERGY
250/306 INSPECTION OF SOLIDS OR LIQUIDS BY CHARGED PARTICLES

2 250/307 (2 OR, 0 XR)
Class 250 : RADIANT ENERGY
250/306 INSPECTION OF SOLIDS OR LIQUIDS BY CHARGED PARTICLES
250/307 .Methods

2 257/184 (0 OR, 2 XR)
Class 257 : ACTIVE SOLID-STATE DEVICES
257/183 HETEROJUNCTION DEVICE
257/184 .Light responsive structure

2 257/251 (0 OR, 2 XR)
Class 257 : ACTIVE SOLID-STATE DEVICES
257/213 FIELD EFFECT DEVICE
257/215 .Charge transfer device
257/251 ..Substantially incomplete signal charge transfer (e.g., bucket brigade)

2 257/440 (2 OR, 0 XR)
Class 257 : ACTIVE SOLID-STATE DEVICES
257/414 RESPONSIVE TO NON-ELECTRICAL SIGNAL (E.G., CHEMICAL, STRESS, LIGHT, OR MAGNETIC FIE

LD SENSORS)
257/428 .Electromagnetic or particle radiation
257/431 ..Light
257/440 ...With different sensor portions responsive t
o
different wavelengths (e.g., color imager)

2 257/443 (0 OR, 2 XR)
Class 257 : ACTIVE SOLID-STATE DEVICES
257/414 RESPONSIVE TO NON-ELECTRICAL SIGNAL (E.G.,
CHEMICAL, STRESS, LIGHT, OR MAGNETIC FIE
LD SENSORS)
257/428 .Electromagnetic or particle radiation
257/431 ..Light
257/443 ...Matrix or array (e.g., single line arrays)

2 257/E27.159 (0 OR, 2 XR)
Class 257 : ACTIVE SOLID-STATE DEVICES
257/E27.102 .....Read-only memory, ROM, structure (EPO)
257/E27.122 .Including active semiconductor component
sensitive to infrared radiation, light,
or electromagnetic
radiation of a shorter wavelength (EPO)

257/E27.127 ..Device controlled by radiation (EPO)
257/E27.13 ...Imager Including structural or functional
details of the device (EPO)
257/E27.15 ....Charge coupled imager (EPO)
257/E27.159 .....CCD or CID color imager (EPO)

2 324/309 (1 OR, 1 XR)
Class 324 : ELECTRICITY: MEASURING AND TESTING
324/300 PARTICLE PRECESSION RESONANCE
324/307 .Using a nuclear resonance spectrometer system

324/309 ..To obtain localized resonance within a sampl
e

2 345/441 (0 OR, 2 XR)
Class 345 : COMPUTER GRAPHICS PROCESSING, OPERATOR
INTERFACE PROCESSING, AND SELECTIVE VISUAL
DISPLAY
SYSTEMS
345/418 COMPUTER GRAPHICS PROCESSING
345/441 .Shape generating

2 345/442 (0 OR, 2 XR)
Class 345 : COMPUTER GRAPHICS PROCESSING, OPERATOR INTERFACE PROCESSING, AND SELECTIVE VISUAL DISPLAY SYSTEMS
345/418 COMPUTER GRAPHICS PROCESSING
345/441 .Shape generating
345/442 ..Curve

2 345/522 (2 OR, 0 XR)
Class 345 : COMPUTER GRAPHICS PROCESSING, OPERATOR INTERFACE PROCESSING, AND SELECTIVE VISUAL DISPLAY SYSTEMS
345/501 COMPUTER GRAPHIC PROCESSING SYSTEM
345/522 .Graphic command processing

2 345/853 (1 OR, 1 XR)
Class 345 : COMPUTER GRAPHICS PROCESSING, OPERATOR INTERFACE PROCESSING, AND SELECTIVE VISUAL DISPLAY SYSTEMS
345/700 OPERATOR INTERFACE (E.G., GRAPHICAL USER INTERFACE)
345/764 .On-screen workspace or object
345/853 ..Hierarchy or network structure

2 345/854 (0 OR, 2 XR)
Class 345 : COMPUTER GRAPHICS PROCESSING, OPERATOR INTERFACE PROCESSING, AND SELECTIVE VISUAL DISPLAY SYSTEMS
345/700 OPERATOR INTERFACE (E.G., GRAPHICAL USER INTERFACE)
345/764 .On-screen workspace or object
345/853 ..Hierarchy or network structure
345/854 ...Navigation within structure

2 369/283 (0 OR, 2 XR)
Class 369 : DYNAMIC INFORMATION STORAGE OR RETRIEVAL
369/272 STORAGE MEDIUM STRUCTURE
369/283 .Layered (e.g., permanent protective layer)

2 369/288 (0 OR, 2 XR)
Class 369 : DYNAMIC INFORMATION STORAGE OR RETRIEVAL
369/272 STORAGE MEDIUM STRUCTURE
369/288 .Specified material

2 370/472 (0 OR, 2 XR)
Class 370 : MULTIPLEX COMMUNICATIONS
370/464 COMMUNICATION TECHNIQUES FOR INFORMATION CARRIED IN PLURAL CHANNELS
370/465 .Adaptive
370/472 ..Byte length

2 428/64.2 (2 OR, 0 XR)
Class 428 : STOCK MATERIAL OR MISCELLANEOUS ARTICLES
428/64.1 CIRCULAR SHEET OR CIRCULAR BLANK
428/64.2 .Recording medium or carrier

2 428/64.4 (0 OR, 2 XR)
Class 428 : STOCK MATERIAL OR MISCELLANEOUS ARTICLES
428/64.1 CIRCULAR SHEET OR CIRCULAR BLANK
428/64.2 .Recording medium or carrier
428/64.4 ..Optical recording medium or carrier

2 428/64.5 (0 OR, 2 XR)
Class 428 : STOCK MATERIAL OR MISCELLANEOUS ARTICLES
428/64.1 CIRCULAR SHEET OR CIRCULAR BLANK
428/64.2 .Recording medium or carrier
428/64.4 ..Optical recording medium or carrier
428/64.5 ...Tellurium containing

2 428/64.6 (0 OR, 2 XR)
Class 428 : STOCK MATERIAL OR MISCELLANEOUS ARTICLES
428/64.1 CIRCULAR SHEET OR CIRCULAR BLANK
428/64.2 .Recording medium or carrier
428/64.4 ..Optical recording medium or carrier
428/64.5 ...Tellurium containing
428/64.6 ....Protective layer

2 428/913 (0 OR, 2 XR)
Class 428 : STOCK MATERIAL OR MISCELLANEOUS ARTICLES
428/913 MATERIAL DESIGNED TO BE RESPONSIVE TO TEMPERATURE, LIGHT, MOISTURE, ETC.

2 430/270.13 (0 OR, 2 XR)
Class 430 : RADIATION IMAGERY CHEMISTRY: PROCESS, COMPOSITION, OR PRODUCT THEREOF
430/269 IMAGING AFFECTING PHYSICAL PROPERTY OF RADIATION SENSITIVE MATERIAL, OR PRODUCING NONPLANAR OR PRINTING SURFACE - PROCESS, COMPOSITION, OR PRODUCT
430/270.1 .Radiation sensitive composition or product or process of making

```
          430/270.11     ..Optical recording nonstructural layered
                              product having a radiation sensitive comp
osition layer
                              claimed or solely disclosed as optically
reorderable and
                              optically machine readable
          430/270.12     ...Having read-write layer of 100 per cent
                            inorganic composition
          430/270.13     ....Which changes phase during recording

  2  430/495.1      (0 OR, 2 XR)
          Class   430 :  RADIATION IMAGERY CHEMISTRY: PROCESS,
                           COMPOSITION, OR PRODUCT THEREOF
          430/495.1      RADIATION SENSITIVE PRODUCT

  2  430/945        (0 OR, 2 XR)
          Class   430 :  RADIATION IMAGERY CHEMISTRY: PROCESS,
                           COMPOSITION, OR PRODUCT THEREOF
          430/945        LASER BEAM

  2  600/420        (2 OR, 0 XR)
          Class   600 :  SURGERY
          600/300        DIAGNOSTIC TESTING
          600/407        .Detecting nuclear, electromagnetic, or
                              ultrasonic radiation
          600/410        ..Magnetic resonance imaging or spectroscopy
          600/420        ...Using detectable material placed in body

  2  607/122        (0 OR, 2 XR)
          Class   607 :  SURGERY: LIGHT, THERMAL, AND ELECTRICAL
                           APPLICATION
          607/1          LIGHT, THERMAL, AND ELECTRICAL APPLICATION
          607/115        .Electrical energy applicator
          607/116        ..Placed in body
          607/119        ...Heart
          607/122        ....Catheter or endocardial (inside heart) typ
e


  2  702/28         (0 OR, 2 XR)
          Class   702 :  DATA PROCESSING: MEASURING, CALIBRATING, OR
                           TESTING
          702/1          MEASUREMENT SYSTEM IN A SPECIFIC ENVIRONMENT
          702/22         .Chemical analysis
          702/27         ..Molecular structure or composition
                              determination
          702/28         ...Using radiant energy
```

```
2  703/5          (1 OR, 1 XR)
        Class   703 :  DATA PROCESSING:  STRUCTURAL DESIGN,
                         MODELING, SIMULATION, AND EMULATION
        703/3          ELECTRICAL ANALOG SIMULATOR
        703/5          .Of physical phenomenon (e.g., heat, wave,
                          geophysics)

2  706/25         (2 OR, 0 XR)
        Class   706 :  DATA PROCESSING: ARTIFICIAL INTELLIGENCE
        706/15         NEURAL NETWORK
        706/25         .Learning method

2  707/10         (1 OR, 1 XR)
        Class   707 :  DATA PROCESSING:  DATABASE AND FILE
                         MANAGEMENT OR DATA STRUCTURES
        707/1          DATABASE OR FILE ACCESSING
        707/10         .Distributed or remote access

2  707/2          (1 OR, 1 XR)
        Class   707 :  DATA PROCESSING:  DATABASE AND FILE
                         MANAGEMENT OR DATA STRUCTURES
        707/1          DATABASE OR FILE ACCESSING
        707/2          .Access augmentation or optimizing

2  707/201        (1 OR, 1 XR)
        Class   707 :  DATA PROCESSING:  DATABASE AND FILE
                         MANAGEMENT OR DATA STRUCTURES
        707/200        FILE OR DATABASE MAINTENANCE
        707/201        .Coherency (e.g., same view to multiple users)


2  709/203        (0 OR, 2 XR)
        Class   709 :  ELECTRICAL COMPUTERS AND DIGITAL PROCESSING
                         SYSTEMS:  MULTIPLE COMPUTER OR PROCESS   COO
RDINATING
        709/200        MULTICOMPUTER DATA TRANSFERRING
        709/201        .Distributed data processing
        709/203        ..Client/server

2  709/236        (1 OR, 1 XR)
        Class   709 :  ELECTRICAL COMPUTERS AND DIGITAL PROCESSING
                         SYSTEMS:  MULTIPLE COMPUTER OR PROCESS   COO
RDINATING
        709/200        MULTICOMPUTER DATA TRANSFERRING
        709/230        .Computer-to-computer protocol implementing
        709/236        ..Computer-to-computer data framing

2  712/11         (0 OR, 2 XR)
```

Class 712 : ELECTRICAL COMPUTERS AND DIGITAL PROCESSING
SYSTEMS: PROCESSING ARCHITECTURES AND INSTRUCTION
PROCESSING
712/1 PROCESSING ARCHITECTURE
712/10 .Array processor
712/11 ..Array processor element interconnection

2 712/20 (2 OR, 0 XR)
Class 712 : ELECTRICAL COMPUTERS AND DIGITAL PROCESSING
SYSTEMS: PROCESSING ARCHITECTURES AND INSTRUCTION
PROCESSING
712/1 PROCESSING ARCHITECTURE
712/10 .Array processor
712/16 ..Array processor operation
712/20 ...Multimode (e.g., MIMD to SIMD, etc.)

2 712/209 (0 OR, 2 XR)
Class 712 : ELECTRICAL COMPUTERS AND DIGITAL PROCESSING
SYSTEMS: PROCESSING ARCHITECTURES AND INSTRUCTION
PROCESSING
712/208 INSTRUCTION DECODING (E.G., BY MICROINSTRUCTION, START ADDRESS GENERATOR, HARDWIRED)
712/209 .Decoding instruction to accommodate plural instruction interpretations (e.g., different dialects,
languages, emulation, etc.)

2 712/21 (0 OR, 2 XR)
Class 712 : ELECTRICAL COMPUTERS AND DIGITAL PROCESSING
SYSTEMS: PROCESSING ARCHITECTURES AND INSTRUCTION
PROCESSING
712/1 PROCESSING ARCHITECTURE
712/10 .Array processor
712/16 ..Array processor operation
712/21 ...Multiple instruction, Multiple data (MIMD)

2 712/22 (0 OR, 2 XR)
Class 712 : ELECTRICAL COMPUTERS AND DIGITAL PROCESSING
SYSTEMS: PROCESSING ARCHITECTURES AND INSTRUCTION
PROCESSING
712/1 PROCESSING ARCHITECTURE

712/10 .Array processor
712/16 ..Array processor operation
712/22 ...Single instruction, multiple data (SIMD)

2 715/500.1 (0 OR, 2 XR)
Class 715 : DATA PROCESSING: PRESENTATION PROCESSING OF DOCUMENT
715/500.1 SYNCHRONIZATION OF DIVERSE MEDIA

2 715/512 (1 OR, 1 XR)
Class 715 : DATA PROCESSING: PRESENTATION PROCESSING OF DOCUMENT
715/512 ANNOTATION CONTROL

2 715/514 (0 OR, 2 XR)
Class 715 : DATA PROCESSING: PRESENTATION PROCESSING OF DOCUMENT
715/514 HIERARCHICAL CONTROL

2 716/18 (1 OR, 1 XR)
Class 716 : DATA PROCESSING: DESIGN AND ANALYSIS OF CIRCUIT OR SEMICONDUCTOR MASK
716/1 CIRCUIT DESIGN
716/18 .Logical circuit synthesizer

2 717/107 (2 OR, 0 XR)
Class 717 : DATA PROCESSING: SOFTWARE DEVELOPMENT, INSTALLATION, AND MANAGEMENT
717/100 SOFTWARE PROGRAM DEVELOPMENT TOOL (E.G., INTEGRATED CASE TOOL OR STAND-ALONE DEVEL OPMENT TOOL)
717/106 .Code generation
717/107 ..Component based

2 717/121 (2 OR, 0 XR)
Class 717 : DATA PROCESSING: SOFTWARE DEVELOPMENT, INSTALLATION, AND MANAGEMENT
717/100 SOFTWARE PROGRAM DEVELOPMENT TOOL (E.G., INTEGRATED CASE TOOL OR STAND-ALONE DEVEL OPMENT TOOL)
717/120 .Managing software components
717/121 ..Software configuration

```
File   6:NTIS 1964-2004/Jun W2
         (c) 2004 NTIS, Intl Cpyrght All Rights Res
File   2:INSPEC 1969-2004/Jun W1
         (c) 2004 Institution of Electrical Engineers
File   8:Ei Compendex(R) 1970-2004/Jun W1
         (c) 2004 Elsevier Eng.  Info. Inc.
File  34:SciSearch(R) Cited Ref Sci 1990-2004/Jun W1
         (c) 2004 Inst for Sci Info
File  35:Dissertation Abs Online 1861-2004/May
         (c) 2004 ProQuest Info&Learning
File  65:Inside Conferences 1993-2004/Jun W2
         (c) 2004 BLDSC all rts. reserv.
File  94:JICST-EPlus 1985-2004/May W4
         (c)2004 Japan Science and Tech Corp(JST)
File  95:TEME-Technology & Management 1989-2004/May W4
         (c) 2004 FIZ TECHNIK
File  99:Wilson Appl. Sci & Tech Abs 1983-2004/May
         (c) 2004 The HW Wilson Co.
File 111:TGG Natl.Newspaper Index(SM) 1979-2004/Jun 15
         (c) 2004 The Gale Group
File 144:Pascal 1973-2004/Jun W1
         (c) 2004 INIST/CNRS
File 202:Info. Sci. & Tech. Abs. 1966-2004/May 14
         (c) 2004 EBSCO Publishing
File 233:Internet & Personal Comp. Abs. 1981-2003/Sep
         (c) 2003 EBSCO Pub.
File 266:FEDRIP 2004/Apr
         Comp & dist by NTIS, Intl Copyright All Rights Res
File 434:SciSearch(R) Cited Ref Sci 1974-1989/Dec
         (c) 1998 Inst for Sci Info
File 483:Newspaper Abs Daily 1986-2004/Jun 11
         (c) 2004 ProQuest Info&Learning
File 583:Gale Group Globalbase(TM) 1986-2002/Dec 13
         (c) 2002 The Gale Group
File 603:Newspaper Abstracts 1984-1988
         (c)2001 ProQuest Info&Learning

Set     Items   Description
S1       4401   PREFETCH? OR PRE()FETCH?
S2     867965   DISTANCE? ?
S3      84186   S2(3N)(DETERMIN? OR DET? ? OR CALCULAT? OR COMPUT??? ?
OR
             COMPUTAT? OR MEASUR???????? ? OR GAUG??? ? ? OR ASSESS? OR
APPRAIS?)
S4      24955   S2(3N)(ANALYS? OR ANALYZ? OR ANALYT? OR QUANTIF? OR
QUANTITAT? OR
             DERIV? OR GENERAT???? ? OR MENSUR? OR EVALUAT? OR QUANTIS?
OR QUANTIZ?)
S5     196791   PIPELIN? OR PIPE()LIN???? ?
S6     106205   TRAVERS???? ?
S7         25   S1(2N)S2
S8         40   S1(10N)S2
S9         10   S8 AND S3:S4
S10         6   RD (unique items)
S11    498663   LIBRARY? OR LIBRARIES
```

```
S12     965203   OBJECT? ? OR OBJECTORIENT? OR OO OR OOP? ? OR OOP?? ?
OR OOF OR
              OOPLA OR OOPL OR ODBM? ? OR OOAD
S13       3616   OODB OR ODMS OR OOL OR OOOS OR OOPS OR OOT OR OOS
S14       1739   LINKED()LIST? ? OR SUBLIST?
S15    2972045   TREE OR TREES OR FOREST? ? OR SUBTREE? ? OR CHILDREN OR
CHILD OR
              NODE OR NODES OR SUBNODE? OR TREEMAP? OR LEAVES OR LEAF? ?
S16     537545   PARENT? ? OR OFFSPRING? OR OFF()SPRING? ? OR ANCEST?R?
? OR DESCEND?NT? ?
S17     539169   PYRAMID? OR CASCAD??? ? OR HIERARCH?
S18        381   S1(7N)S11:S17
S19       3258   S6(7N)S11:S17
S20         13   S18 AND S19
S21         17   S18:S19(20N)S5
S22          8  S21 AND S1
S23         16  (S20 OR S22) NOT S9
S24          9  RD (unique items)
```

24/7/1 (Item 1 from file: 6)
DIALOG(R)File 6:NTIS

2193856 NTIS Accession Number: ADA387165/XAB
Improving Index Performance through Prefetching
Chen, S. ; Gibbons, P. B. ; Mowry, T. C.
Carnegie-Mellon Univ., Pittsburgh, PA. School of Computer Science.
Corp. Source Codes: 005343049; 423887
Report No.: CMU-CS-00-177
Dec 2000 28p
Languages: English
Journal Announcement: USGRDR0113
Product reproduced from digital image. Order this product from NTIS by: phone at 1-800-553-NTIS (U.S. customers); (703)605-6000 (other countries); fax at (703)605-6900; and email at orders@ntis.gov. NTIS is located at 5285 Port Royal Road, Springfield, VA, 22161, USA.
NTIS Prices: PC A03/MF A01
Country of Publication: United States
In recognition of the crucial role that cache hierarchies play in database performance, recent studies have revisited core database algorithms and data structures in an effort to reduce the number of cache misses. While these efforts to avoid cache misses are certainly helpful they are not a complete solution for two reasons. First, a large number of cache misses still remain that cannot be eliminated. Second, because modern processors support prefetching and other mechanisms to potentially overlap cache misses with computation and other misses, it is not the total number of cache misses that dictates performance, but rather the total amount of exposed miss latency. Hence an algorithm that is more amenable to prefetching can potentially out perform an algorithm with fewer cache misses. In this paper, we propose and evaluate **Prefetching** B+ **Trees** (pB+ **Trees**). Such **trees** are designed to exploit **prefetching** to accelerate two important operations on B+ **Tree** indices: searches and range scans. To accelerate searches, pB+ **Trees** use **prefetching** to effectively create wider **nodes** than the natural data transfer size: e.g., eight vs. one cache lines or disk pages. These wider nodes reduce the height of the B+ Tree, thereby decreasing the number of expensive misses when going from parent to child without significantly increasing the cost of fetching a given node. Our results show that this technique speeds up search, insertion, and deletion times by a factor of 1.2-1.5 for main-memory B+ Trees. In addition, it outperforms and is complimentary to Cache-Sensitive B+ Trees. To accelerate range scans, pB+ Trees provide arrays of pointers to their **leaf nodes** . These allow the pB+ **Tree** to **prefetch** arbitrarily far ahead, even for nonclustered indices, thereby hiding the normally expensive cache misses associated with **traversing** the **leaves** within the range. Our results show that this technique yields over a sixfold speedup on range scans of over 1000 keys.

24/7/2 (Item 1 from file: 2)
DIALOG(R)File 2:INSPEC

7387429 INSPEC Abstract Number: C2002-10-6120-023
Title: A programmable memory hierarchy **for** prefetching **linked data structures**
Author(s): Chia-Lin Yang; Lebeck, A.
Author Affiliation: Nat. Taiwan Univ., Taipei, Taiwan
Conference Title: High Performance Computing. 4th International Symposium, ISHPC 2002. Proceedings (Lecture Notes in Computer Science

Vol.2327) p.160-74
Editor(s): Zima, H.P.; Joe, K.; Sato, M.; Seo, Y.; Shimasaki, M.
Publisher: Springer-Verlag, Berlin, Germany
Publication Date: 2002 Country of Publication: Germany xv+564 pp.
ISBN: 3 540 43674 X Material Identity Number: XX-2002-01484
Conference Title: High Performance Computing. 4th International Symposium, ISHPC 2002. Proceedings
Conference Date: 15-17 May 2002 Conference Location: Kansai Science City, Japan
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: Prefetching is often used to overlap memory latency with computation for array-based applications. However, prefetching for pointer-intensive applications remains a challenge because of the irregular memory access pattern and pointer-chasing problem. In this paper, we use a programmable processor, a prefetch engine (PFE), at each level of the memory **hierarchy** to cooperatively execute instructions that **traverse** a linked data structure. Cache blocks accessed by the processors at the L2 and memory levels are proactively pushed up to the CPU. We look at several design issues to support this programmable memory hierarchy. We establish a general interaction scheme among three PFEs and design a mechanism to synchronize the PFE execution with the CPU. Our simulation results show that the proposed prefetching scheme can reduce up to 100% of memory stall time on a suite of pointer-intensive applications, reducing overall execution time by an average 19%. (38 Refs)
Subfile: C

24/7/3 (Item 2 from file: 2)
DIALOG(R)File 2:INSPEC

7128590 INSPEC Abstract Number: C2002-01-6150C-043
Title: Speculative prefetching of induction pointers
Author(s): Stoutchinin, A.; Amaral, J.N.; Gao, G.R.; Dehnert, J.C.; Jain, S.; Douillet, A.
Author Affiliation: STMicroelectronics, Grenoble, France
Conference Title: Compiler Construction. 10th International Conference, CC 2001. Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2001. Proceedings (Lecture Notes in Computer Science Vol.2027) p.289-303
Editor(s): Wilhelm, R.
Publisher: Springer-Verla, Berlin, Germany
Publication Date: 2001 Country of Publication: Germany xi+370 pp.
ISBN: 3 540 41861 X Material Identity Number: XX-2001-01778
Conference Title: Compiler Construction. 10th International Conference
Conference Sponsor: ELSAG; Fondazione Cassa di Risparmio di Genova e Imperia; INDAM - Gruppo Nazionale per l'Informatica Matematica; et al
Conference Date: 2-6 April 2001 Conference Location: Genova, Italy
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: We present an automatic approach for **prefetching** data for **linked list** data structures. The main idea is based on the observation that linked list elements are frequently allocated at constant distance from one another in the heap. When **linked lists** are **traversed** , a regular pattern of memory accesses with constant stride emerges. This regularity in the memory footprint of **linked lists** enables the development of a **prefetching** framework where the address of the element accessed in one of the future iterations of the loop is dynamically predicted based on its previous regular behavior. We automatically identify

pointer-chasing recurrences in loops that access linked lists. This identification uses a surprisingly simple method that looks for induction pointers: pointers that are updated in each loop iteration by a load with a constant offset. We integrate induction pointer prefetching with loop scheduling. A key intuition incorporated in our framework is to insert prefetches only if there are processor resources and memory bandwidth available. In order to estimate available memory bandwidth, we calculate the number of potential cache misses in one loop iteration. Our estimation algorithm is based on an application of graph coloring on a memory access interference graph derived from the control flow graph. We implemented the prefetching framework in an industry-strength production compiler, and performed experiments on ten benchmark programs with linked lists. We observed performance improvements between 15% and 35% in three of them. (20 Refs)

Subfile: C

24/7/4 (Item 3 from file: 2)

DIALOG(R)File 2:INSPEC

6498665 INSPEC Abstract Number: C2000-03-6120-032

Title: A prefetching technique for irregular accesses to linked data structures

Author(s): Karlsson, M.; Dahlgren, F.; Stenstrom, P.

Author Affiliation: Dept. of Comput. Eng., Chalmers Univ. of Technol., Goteborg, Sweden

Conference Title: Proceedings Sixth International Symposium on High-Performance Computer Architecture. HPCA-6 (Cat. No.PR00550) p. 206-17

Publisher: IEEE Comput. Soc, Los Alamitos, CA, USA

Publication Date: 1999 Country of Publication: USA xiii+420 pp.

ISBN: 0 7695 0550 3 Material Identity Number: XX-2000-00100

U.S. Copyright Clearance Center Code: 0 7695 0550 3/2000/$10.00

Conference Title: Proceedings of HPCA: 6th International Symposium on High-Performance Computer Architecutre

Conference Sponsor: IEEE Comput. Soc. Tech. Committee on Comput. Archit

Conference Date: 8-12 Jan. 2000 Conference Location: Touluse, France

Language: English Document Type: Conference Paper (PA)

Treatment: Practical (P)

Abstract: Prefetching offers the potential to improve the performance of linked data structure (LDS) traversals. However, previously proposed prefetching methods only work well when there is enough work processing a **node** that the **prefetch** latency can be hidden, or when the LDS is long enough and the traversal path is known a priori. This paper presents a prefetching technique called prefetch arrays which can prefetch both short LDS, as the lists found in hash tables, and **trees** when the **traversal** path is nor known a priori. We offer two implementations, one software-only and one which combines software annotations with a prefetch engine in hardware. On a pointer-intensive benchmark suite, we show that our implementations reduce the memory stall lime by 23% to 51% for the kernels with **linked lists** , while the other **prefetching** methods cause reductions that are substantially less. For binary-trees, our hardware method manages to cut nearly 60% of the memory stall time even when the traversal path is not known a priori. However, when the branching factor of the tree is too high, our technique does not improve performance. Another contribution of the paper is that we quantify pointer-chasing found in interesting applications such as OLTP, Expert Systems, DSS, and JAVA codes and discuss which prefetching techniques are relevant to use in each case. (20 Refs)

Subfile: C
Copyright 2000, IEE

24/7/5 (Item 4 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

5162034 INSPEC Abstract Number: C9602-5220-007
Title: Proceedings of MICRO`95: 28th Annual IEEE/ACM International Symposium on Microarchitecture
Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA
Publication Date: 1995 Country of Publication: USA xiv+361 pp.
ISBN: 0 8186 7351 6 Material Identity Number: XX95-02583
U.S. Copyright Clearance Center Code: 95/$4.00
Conference Title: Proceedings of MICRO`95: 28th Annual IEEE/ACM International Symposium on Microarchitecture
Conference Sponsor: IEEE Tech. Committee on Microprogramming & Microarchit.; ACM - SIGMICRO
Conference Date: 29 Nov.-1 Dec. 1995 Conference Location: Ann Arbor, MI, USA
Language: English Document Type: Conference Proceedings (CP)
Abstract: The following topics were dealt with: branch prediction; instruction-level parallel compilation; memory system issues; software **pipelining** ; data flow and multithreading architectures; analysis of branching architecture; software and hardware **object** code translation; data **prefetching** ; design direction for future processors; multithreading and superscalar design studies; and architectural features for instruction-level parallelism.
Subfile: C
Copyright 1996, IEE

24/7/6 (Item 1 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

04346122 E.I. No: EIP96023024056
Title: Proceedings of the 1995 28th Annual International Symposium on Microarchitecture
Author: Anon (Ed.)
Conference Title: Proceedings of the 1995 28th Annual International Symposium on Microarchitecture
Conference Location: Ann Arbor, MI, USA Conference Date: 19951129-19951201
Sponsor: IEEE; Association for Computing Machinery - SIGMICRO
E.I. Conference No.: 44295
Source: Proceedings of the Annual International Symposium on Microarchitecture 1995. IEEE, Los Alamitos, CA, USA. 360p
Publication Year: 1995
CODEN: PSMIE7 ISSN: 1072-4451
Language: English
Document Type: CP; (Conference Proceedings) Treatment: A; (Applications)
Journal Announcement: 9604W1
Abstract: The proceedings contains 37 papers. Topics discussed include performance of branch correlation schemes, instruction level parallelism compilation, computer memory systems, software **pipelining** , data flow and multithreading architectures, branch architecture analysis, software and hardware **object** code translation, data **prefetching** , multithreading and

superscalar design studies.

24/7/7 (Item 1 from file: 94)
DIALOG(R)File 94:JICST-EPlus

04655967 JICST ACCESSION NUMBER: 00A0799949 FILE SEGMENT: JICST-E
A Data Prefeching Mechanism for A Linked List Structure.
YAMAMURA SHUJI (1); HIRATA HIROAKI (1); SHIBAYAMA KIYOSHI (1); NIIMI HARUO (2)
(1) Kyoto Inst. of Technol., Fac. of Eng. and Des.; (2) Engineering, Kyoto Sangyo Univ.
Joho Shori Gakkai Shinpojiumu Ronbunshu, 2000, VOL.2000,NO.6, PAGE.115-122, FIG.6, TBL.3, REF.8
JOURNAL NUMBER: Y0978BAT ISSN NO: 1344-0640
UNIVERSAL DECIMAL CLASSIFICATION: 681.326
LANGUAGE: Japanese COUNTRY OF PUBLICATION: Japan
DOCUMENT TYPE: Conference Proceeding
ARTICLE TYPE: Original paper
MEDIA TYPE: Printed Publication
ABSTRACT: In this paper, we propose a novel data **prefetching** scheme for a **linked list** data structure. Most of existing data **prefetch** techniques are designed for array structures, and efficient techniques for linked lists have not been developed well. In our scheme, a processor **traverses** the **linked list** structure automatically and preloads the pointer values, in order to hide data load latency as well as the memory access latency. Our simulation result of this scheme achieves 10.7% improvement in the execution performance. (author abst.)

24/7/8 (Item 1 from file: 144)
DIALOG(R)File 144:Pascal

16140025 PASCAL No.: 03-0293517
A programmable memory hierarchy **for** prefetching **linked data structures**
High performance computing : Kansai Science City, 15-17 May 2002
YANG Chia-Lin; LEBECK Alvin
ZIMA Hans P, ed; JOE Kazuki, ed; SATO Mitsuhisa, ed; SEO Yoshiki, ed; SHIMASAKI Masaaki, ed
National Taiwan University, 1 Roosevelt Rd. Sec. 4, Taipei, Taiwan; Duke University, Durham, NC 27708, United States
ISHPC 2002 : international symposium on high performance computing, 4 (Kansai Science City JPN) 2002-05-15
Journal: Lecture notes in computer science, 2002, 2327 160-174
ISBN: 3-540-43674-X ISSN: 0302-9743 Availability: INIST-16343; 354000108473830130
No. of Refs.: 38 ref.
Document Type: P (Serial); C (Conference Proceedings) ; A (Analytic)
Country of Publication: Germany
Language: English
Prefetching is often used to overlap memory latency with computation for array-based applications. However, prefetching for pointer-intensive applications remains a challenge because of the irregular memory access pattern and pointer-chasing problem. In this paper, we use a programmable processor, a prefetch engine (PFE), at each level of the memory **hierarchy** to cooperatively execute instructions that **traverse** a linked data structure. Cache blocks accessed by the processors at the L2 and memory

levels are proactively pushed up to the CPU. We look at several design issues to support this programmable memory hierarchy. We establish a general interaction scheme among three PFEs and design a mechanism to synchronize the PFE execution with the CPU. Our simulation results show that the proposed prefetching scheme can reduce up to 100% of memory stall time on a suite of pointer-intensive applications, reducing overall execution time by an average 19%.

24/9/9 (Item 2 from file: 144)
DIALOG(R)File 144:Pascal

15793636 PASCAL No.: 02-0508800
MULTIFLOT SIMULTANE ET MULTIMEDIA : OUTIL, ETUDE ET OPTIMISATION (Simultaneous Multithreading and Multimedia : Tool, Study and Optimization)
LIMOUSIN Claude; TEMAM Olivier, dir
Universite de Paris 11, Orsay, France
Univ.: Universite de Paris 11. Orsay. FRA Degree: Th. doct.
2001-10; 2001 159 p.
Availability: INIST-T 139981; T01PA112169 0000; RBCCN-914712101; T01PA112169 0000
No. of Refs.: 123 ref.
Document Type: T (Thesis) ; M (Monographic)
Country of Publication: France
Language: French Summary Language: French; English
Du fait du manque de parallelisme d'instructions et de l'ecart toujours croissant entre la vitesse des processeurs et le temps d'acces de la memoire dynamique, un unique flot d'instructions ne peut utiliser efficacement le materiel maintenant integrable dans un processeur. Une solution est d'executer plusieurs flots d'instructions a travers le multiflot simultane (SMT), qui permet a tous les flots presents sur le processeur d'acceder simultanement a toutes les unites d'execution, est une approche prometteuse. Dans cette these, nous etudions les performances d'applications multimedias sur cette architecture. Ces applications, en plein developpement actuellement, possedent des caracteristiques particulieres qui font du SMT une plate-forme interessante pour les executer. Dans un premier temps, nous presentons PopSPY, un instrumenteur de code executable pour le PowerPC. PopSPY permet notamment la generation de traces multiflot de programmes dont les sources ne sont pas disponibles, comme certains programmes utilises dans cette these. Il permet egalement la generation de traces AltiVec, l'extension multimedia du PowerPC. Nous evaluons ensuite les performances de l'architecture SMT sur nos applications en considerant differents dimensionnements du coeur superscalaire et de la hierarchie memoire. Nous montrons que le SMT permet de masquer la latence du cache de second niveau et est efficace sur ces applications, meme avec un grand nombre de flots. Les jeux d'instructions multimedia presents dans les processeurs ainsi que le prechargement de donnees sont deux moyens d'ameliorer les performances de ces applications. Nous analysons le benefice de l'association de ces deux mecanismes avec le SMT, a la fois separement et conjointement, en fonction des caracteristiques de la hierarchie memoire. Nous montrons notamment que l'utilisation d'unites multimedia se justifie de moins en moins a mesure que le nombre de flots augmente ou que le d'ebit memoire diminue. Enfin, nous nous penchons sur l'optimisation particuliere du multiflot simultane pour la 3D polygonale temps-reel en tirant parti de la structure du pipeline graphique qui constitue le coeur de cette application. Nous mettons a jour une pollution des stations de reservation du fait des echecs dans le cache de second niveau, et proposons un mecanisme de prediction

pour controler l'emission des flots et eviter cette pollution. Nous obtenons ainsi des gains de performance significatifs.

English Descriptors: Storage management; **Hierarchy** ; Multithread; Multimedia; **Pipeline** processing; Cache memory; Data **prefetching**

French Descriptors: Gestion memoire; Hierarchie; Multitache; Multimedia; Traitement pipeline; Antememoire; Prechargement donnee

Classification Codes: 001D02B07C

?

10/7/1 (Item 1 from file: 2)
DIALOG(R)File 2:INSPEC

6987293 INSPEC Abstract Number: C2001-09-6110P-003
Title: Adaptive software prefetching in scalable multiprocessors using cache information
Author(s): Daeyeon Park; Byeong Hag Seong; Saavedra, R.H.
Author Affiliation: Dept. of Electr. Eng., Korea Adv. Inst. of Sci. & Technol., Seoul, South Korea
Journal: Parallel Computing vol.27, no.9 p.1173-95
Publisher: Elsevier,
Publication Date: Aug. 2001 Country of Publication: Netherlands
CODEN: PACOEJ ISSN: 0167-8191
SICI: 0167-8191(200108)27:9L.1173:ASPS;1-Y
Material Identity Number: F777-2001-008
U.S. Copyright Clearance Center Code: 0167-8191/2001/$20.00
Document Number: S0167-8191(01)00085-0
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P)
Abstract: Scalable multiprocessors present special challenges to static software prefetching because on these systems the memory access latency is not completely determined at compile time. Furthermore, dynamic software prefetching cannot do much better because individual nodes on large-scale multiprocessors would tend to experience different remote memory delays over time. A fixed **prefetch distance** , even when **computed** at run-time, cannot perform well for the whole duration of a software pipeline. Here we present an adaptive scheme for software prefetching that makes it possible for nodes to dynamically change, not only the amount of **prefetching** , but the **prefetch distance** as well. We show how simple performance data collected by hardware monitors can allow programs to observe, evaluate and change their prefetching policies. Our results show that adaptive prefetching (APF) was capable of improving performance over static and dynamic prefetching by 10%-60%. (12 Refs)
Subfile: C

10/7/2 (Item 2 from file: 2)
DIALOG(R)File 2:INSPEC

5425593 INSPEC Abstract Number: C9612-5440-016
Title: Improving the effectiveness of software prefetching with adaptive executions
Author(s): Saavedra, R.H.; Daeyeon Park
Author Affiliation: Dept. of Comput. Sci., Univ. of Southern California, Los Angeles, CA, USA
Conference Title: Proceedings of the 1996 Conference on Parallel Architectures and Compilation Techniques (PACT '96) (Cat. No.96TB100077) p.68-78
Publisher: IEEE Comput. Soc. Press, Los Alamitos, CA, USA
Publication Date: 1996 Country of Publication: USA xiv+304 pp.
ISBN: 0 8186 7632 9 Material Identity Number: XX96-03010
U.S. Copyright Clearance Center Code: 1089-795X/96/$5.00
Conference Title: Proceedings of PACT '96: International Conference on Parallel Architectures and Compilation Techniques
Conference Sponsor: IEEE Comput. Soc. Tech. Committee on Comput. Archit.; IFIP WG 10.3 (Concurrent Syst.); ACM SIGARCH
Conference Date: 20-23 Oct. 1996 Conference Location: Boston, MA, USA

Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: The effectiveness of software prefetching for tolerating latency depends mainly on the ability of programmers and/or compilers to: 1) predict in advance the magnitude of the run-time remote memory latency, and 2) insert **prefetches** at a **distance** that minimizes stall time without causing cache pollution. Scalable heterogeneous multiprocessors, such as network of computers (NOWs), present special challenges to static software prefetching because on these systems the network topology and node configuration are not completely determined at compile time. Furthermore, dynamic software prefetching cannot do much better because individual nodes on heterogeneous large NOWs would tend to experience different remote memory delays over time. A fixed **prefetch distance** , even when **computed** at run-time, cannot perform well for the whole duration of a software pipeline. Here we present an adaptive scheme for software prefetching that makes it possible for nodes to dynamically change, not only the amount of **pref**etching , but the **prefetch distance** as well. Doing this makes it possible to tailor the execution of software pipeline to the prevailing conditions affecting each node. We show how simple performance data collected by hardware monitors can allow programs to observe, evaluate and change their prefetching policies. Our results show that on the benchmarks we simulated adaptive prefetching was capable of improving performance over static and dynamic prefetching by 10% to 60%. More important, future increases in the heterogeneity and size of NOWs will increase the advantages of adaptive prefetching over static and dynamic schemes. (16 Refs)
Subfile: C

10/7/3 (Item 3 from file: 2)
DIALOG(R)File 2:INSPEC

5006744 INSPEC Abstract Number: C9509-6120-013
Title: Effective cache prefetching on bus-based multiprocessors
Author(s): Tullsen, D.M.; Eggers, S.J.
Author Affiliation: Dept. of Comput. Sci. & Eng., Washington Univ., Seattle, WA, USA
Journal: ACM Transactions on Computer Systems vol.13, no.1 p.57-88
Publication Date: Feb. 1995 Country of Publication: USA
CODEN: ACSYEC ISSN: 0734-2071
U.S. Copyright Clearance Center Code: 0734-2071/95/0200-0057$03.50
Language: English Document Type: Journal Paper (JP)
Treatment: Theoretical (T)
Abstract: Compiler-directed cache prefetching has the potential to hide much of the high memory latency seen by current and future high-performance processors. However, prefetching is not without costs, particularly on a shared-memory multiprocessor. Prefetching can negatively affect bus utilization, overall cache miss rates, memory latencies and data sharing. We simulate the effects of a compiler-directed prefetching algorithm, running on a range of bus-based multiprocessors. We show that, despite a high memory latency, this architecture does not necessarily support prefetching well, in some cases actually causing performance degradations. We pinpoint several problems with prefetching on a shared-memory architecture (additional conflict misses, no reduction in the data-sharing traffic; and associated latencies, a multiprocessor's greater sensitivity to memory utilization and the sensitivity of the cache hit rate to **prefetch distance**) and **measure** their effect on performance. We then solve those problems through architectural techniques and heuristics for prefetching that could be easily incorporated into a compiler: (1) victim

caching, which eliminates most of the cache conflict misses caused by prefetching in a direct-mapped cache, (2) special prefetch algorithms for shared data, which significantly improve the ability of our basic prefetching algorithm to prefetch invalidation misses, and (3) compiler based shared-data restructuring, which eliminates many of the invalidation misses the basic prefetching algorithm does not predict. The combined effect of these improvements is to make prefetching effective over a much wider range of memory architectures. (32 Refs)
Subfile: C

10/7/4 (Item 1 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

06138713 E.I. No: EIP02397099189
Title: Scalable data management using user-based caching and prefetching in distributed virtual environments
Author: Park, Sungju; Lee, Dongman; Lim, Mingyu; Yu, Chansu
Corporate Source: Info. and Communications University, Taejon 305-348, South Korea
Conference Title: Proceedings of the ACM Symposium on Virtual Reality Software and Technology (VRST 2001)
Conference Location: Banff, Alta., Canada Conference Date: 20011115-20011117
Sponsor: ACM; SIGGRAPH; SIGCHI
E.I. Conference No.: 59601
Source: ACM Symposium on Virtual Reality Software and Technology, Proceedings, VRST 2001. p 121-126
Publication Year: 2001
Language: English
Document Type: CA; (Conference Article) Treatment: T; (Theoretical); X; (Experimental)
Journal Announcement: 0209W5
Abstract: For supporting real-time interaction in distributed virtual environments (DVEs), it is common to replicate virtual world data at clients from the server. For efficient replication, two schemes are used together in general - prioritized transfer of objects and a caching and prefetching technique. Existing caching and **prefetching** approaches for DVEs exploit spatial relationship based on **distances** between a user and objects. However, spatial relationship fails to determine which types of objects are more important to an individual user, not reflecting user's interests. We propose a scalable data management scheme using user-based caching and **prefetching** exploiting the object's access priority **generated** from spatial **distance** and individual user's interest in objects in DVEs. We also further improve the cache hit rate by incorporating user's navigation behavior into the spatial relationship between a user and the objects in the cache. By combining the interest score and popularity score of an object with the spatial relationship, we improve the performance of caching and prefetching since the interaction locality between the user and objects are reflected in addition to spatial locality. The simulation results show that the proposed scheme outperforms the hit rate of existing caching and prefetching by 10% on average when the cache size is set to basic cache size, the size of expected number of objects included in the user's viewing range. 15 Refs.

10/7/5 (Item 2 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

05855324 E.I. No: EIP01296588911
Title: Adaptive software prefetching in scalable multiprocessors using cache information
Author: Park, D.; Byeong Hag Seong; Saavedra, R.H.
Corporate Source: Department of Electrical Engineering Korea Advanced Inst. of Sci./Tech., Taejon 305-701, South Korea
Source: Parallel Computing v 27 n 9 August 2001. p 1173-1195
Publication Year: 2001
CODEN: PACOEJ ISSN: 0167-8191
Language: English
Document Type: JA; (Journal Article) Treatment: A; (Applications); T; (Theoretical)
Journal Announcement: 0107W4
Abstract: Scalable multiprocessors present special challenges to static software prefetching because on these systems the memory access latency is not completely determined at compile time. Furthermore, dynamic software prefetching cannot do much better because individual nodes on large-scale multiprocessors would tend to experience different remote memory delays over time. A fixed **prefetch distance** , even when **computed** at run-time, cannot perform well for the whole duration of a software pipeline. Here we present an adaptive scheme for software prefetching that makes it possible for nodes to dynamically change, not only the amount of **prefetching** , but the **prefetch distance** as well. We show how simple performance data collected by hardware monitors can allow programs to observe, evaluate and change their prefetching policies. Our results show that adaptive prefetching (APF) was capable of improving performance over static and dynamic prefetching by 10%-60%. copy 2001 Elsevier Science B.V. 12 Refs.

10/7/6 (Item 1 from file: 34)
DIALOG(R)File 34:SciSearch(R) Cited Ref Sci

03463176 Genuine Article#: PG306 Number of References: 30
Title: PERFORMANCE AND OPTIMIZATION OF DATA PREFETCHING STRATEGIES IN SCALABLE MULTIPROCESSORS
Author(s): SAAVEDRA RH; MAO WH; HWANG K
Corporate Source: UNIV SO CALIF,DEPT COMP SCI/LOS ANGELES//CA/90089; UNIV SO CALIF,DEPT ELECT ENGN/LOS ANGELES//CA/90089
Journal: JOURNAL OF PARALLEL AND DISTRIBUTED COMPUTING, 1994, V22, N3 (SEP) , P427-448
ISSN: 0743-7315
Language: ENGLISH Document Type: ARTICLE
Abstract: Prefetching is one of several techniques for hiding and tolerating the large memory latencies of scalable multiprocessors. In this paper, we present a performance model for analyzing the limits and effectiveness of data prefetching. The model incorporates the effects of program behavior, network characteristics, cache coherency protocols, and memory consistency model. Our results indicate that, as long as there is enough extra network bandwidth, prefetching is very effective in hiding large latencies. In machines with sufficiently large caches to hold the program working set, the intra- and internode cache interference is marginally low enough to have any significant impact on **prefetching** performance. Furthermore, we reveal the fact that the effective **prefetch distance** plays a vital role and adapts extremely well to changes in cache miss rates and remote latencies, thus allowing prefetches to be more effective in hiding latency. An adaptive algorithm is provided to optimize the **prefetch distance** .

This is based on the dynamic behavior of the application, interconnection network, and distributed caches and memories. This optimization of the **prefetch distance** constitutes a significant advantage of **prefetching** over other latency tolerating techniques, such as multithreading. We show that the **prefetch distance** can be chosen constant, program-dependent, or decided by performance information. The optimal **distance** could be adaptively **determined** using both compile-time and runtime conditions. Our results are therefore useful not only to compiler writers, but also for the development of runtime support systems in multiprocessors. In large-scale systems, in which network traffic control predominates the performance, the ultimate goal is to match program behavior with machine behavior.

File 348:EUROPEAN PATENTS 1978-2004/Jun W02
(c) 2004 European Patent Office
File 349:PCT FULLTEXT 1979-2002/UB=20040610,UT=20040603
(c) 2004 WIPO/Univentio

Set	Items	Description
S1	2448	PREFETCH? OR PRE()FETCH?
S2	452691	DISTANCE? ?
S3	46673	S2(3N)(DETERMIN? OR DET? ? OR CALCULAT? OR COMPUT??? ? OR COMPUTAT? OR MEASUR???????? ? OR GAUG??? ? ? OR ASSESS? OR APPRAIS?)
S4	7011	S2(3N)(ANALYS? OR ANALYZ? OR ANALYT? OR QUANTIF? OR QUANTITAT? OR DERIV? OR GENERAT???? ? OR MENSUR? OR EVALUAT? OR QUANTIS? OR QUANTIZ?)
S5	25096	PIPELIN? OR PIPE()LIN???? ?
S6	259183	TRAVERS???? ?
S7	9	S1(2N)S2
S8	20	S1(10N)S2
S9	8	S8(25N)S3:S4
S10	8	IDPAT (sorted in duplicate/non-duplicate order)
S11	8	IDPAT (primary/non-duplicate records only)
S12	3592	PIPE()(LINE? ? OR LINING)
S13	1151	S2(3N)(DERIVE? ? OR DERIVING OR DERIVATION?)
S14	0	S8(25N)S13
S15	62431	LIBRARY? OR LIBRARIES
S16	737756	OBJECT? ? OR OBJECTORIENT? OR OO OR OOP? ? OR OOP?? ? OR OOF OR OOPLA OR OOPL OR ODBM? ? OR OOAD
S17	9625	OODB OR ODMS OR OOL OR OOOS OOPS OR OOT OR OOS
S18	3332	LINKED()LIST? ? OR SUBLIST?
S19	228819	TREE OR TREES OR FOREST? ? OR SUBTREE? ? OR CHILDREN OR CHILD OR NODE OR NODES OR SUBNODE? OR TREEMAP? OR LEAVES OR LEAF? ?
S20	69482	PARENT? ? OR OFFSPRING? OR OFF()SPRING? ? OR ANCEST?R? ? OR DESCEND?NT? ?
S21	83377	PYRAMID? OR CASCAD??? ? OR HIERARCH?
S22	162	S1(7N)S15:S21
S23	4512	S6(7N)S15:S21
S24	2	S22(25N)S23
S25	34	S22:S23(20N)(S5 OR S12)
S26	15	S25(25N)S1
S27	17	(S24 OR S26) NOT S11
S28	17	IDPAT (sorted in duplicate/non-duplicate order)
S29	16	IDPAT (primary/non-duplicate records only)

29/5,K/5 **(Item 5 from file: 348)**
DIALOG(R)File 348:EUROPEAN PATENTS

00768818
COMPUTER WITH TWO-DIMENSIONAL MERGE TOURNAMENT SORT USING CACHING
RECHNER MIT ZWEIDIMENSIONALER MISCH-TOURNIER-SORTIERUNG UNTER VERWENDUNG VON CACHESPEICHERUNG
ORDINATEUR A TRI PAR TOURNOI ET FUSIONNEMENT BIDIMENSIONNEL A L'AIDE D'UNE UNITE ANTEMEMOIRE
PATENT ASSIGNEE:
AMDAHL CORPORATION, (628800), 1250 East Arques Avenue, Sunnyvale California 94086, (US), (applicant designated states: DE;GB)
INVENTOR:
COSSOCK, David, 1015 Euclid, Berkeley, CA 94708, (US)
LEGAL REPRESENTATIVE:
Crawford, Andrew Birkby et al (29761), A.A. THORNTON & CO. Northumberland House 303-306 High Holborn, London WC1V 7LE, (GB)
PATENT (CC, No, Kind, Date): EP 782725 A1 970709 (Basic)
EP 782725 B1 980722
WO 9609580 960328
APPLICATION (CC, No, Date): EP 95933072 950918; WO 95US11468 950918
PRIORITY (CC, No, Date): US 308751 940919
DESIGNATED STATES: DE; GB
INTERNATIONAL PATENT CLASS: G06F-007/24;
NOTE:
No A-document published by EPO
LEGAL STATUS (Type, Pub Date, Kind, Text):
Application: 960703 A International application (Art. 158(1))
Application: 970709 A1 Published application (A1with Search Report ;A2without Search Report)
Examination: 970709 A1 Date of filing of request for examination: 970324
Examination: 971112 A1 Date of despatch of first examination report: 970925
Grant: 980722 B1 Granted patent
Oppn None: 990714 B1 No opposition filed
LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	9830	580
CLAIMS B	(German)	9830	556
CLAIMS B	(French)	9830	697
SPEC B	(English)	9830	6668
Total word count - document A			0
Total word count - document B			8501
Total word count - documents A + B			8501

...SPECIFICATION 1/2 + 1/4 + ... + 1/(2k))) nodes (for more than half) are contained in the **leaf subtrees** . Thus a single **prefetch** of codes for **nodes** in a **subtree** will eliminate most misses for a single **ancestor** chain **traversal** , and the **leaf subtrees** can, for example, be stored in the secondary cache, thus increasing the group size n...

...with probability 1/2 and analogously 1/4, 1/8, ... probabilities exist for the remaining **objects** on the path just **traversed** . By **prefetching** the **leaf node subtree** of its **parent** (which is required if the probability prediction is correct), a several-hundred cycle miss penalty ...

...be avoided. In addition, while processing the output list for the

current winner, the required **leaf node** for the next **ancestor** chain **traversal** can be **pre - fetched** .

This ability to 'plan ahead' and thereby make use of prefetch facilities is not available...formation of natural subtrees so that ii.a) fewer cache lines are accessed per tournament **tree traversal** , and ii.b) probabilistic and direct **pre - fetching** of **leaf subtrees** is facilitated by increased cycles per cache line, resulting in fewer pre-fetches per unit...

?

? t11/5,k/all

11/5,K/1 (Item 1 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

01581583
Storage system having means for acquiring execution information of a database management system
Speichersystem mit Mitteln zum Erwerb von Ausfuhrungsinformation eines Datenbanksystems
Systeme de stockage avec moyen d' acquisition de la information d'execution de une base de donnees
PATENT ASSIGNEE:
Hitachi, Ltd., (204144), 6, Kanda Surugadai 4-chome, Chiyoda-ku, Tokyo, (JP), (Applicant designated States: all)
INVENTOR:
Mogi, Kazuhiko Hitachi Ltd. Intellect. Prop. Group, Marunouchi Bldg. 5-1 Marunouchi 1-chome Chiyoda-ku, Tokyo 100-8220, (JP)
Oeda, Takashi Hitachi Ltd. Intell. Prop. Group, Marunouchi Bldg. 5-1 Marunouchi 1-chome Chiyoda-ku, Tokyo 100-8220, (JP)
Kitsuregawa, Masaru, 17, Nijusseikigaoka Maruyamacho, Matsudo-shi Chiba, (JP)
LEGAL REPRESENTATIVE:
Calderbank, Thomas Roger et al (50122), MEWBURN ELLIS York House 23 Kingsway, London WC2B 6HP, (GB)
PATENT (CC, No, Kind, Date): EP 1313034 A2 030521 (Basic)
APPLICATION (CC, No, Date): EP 2002251622 020307;
PRIORITY (CC, No, Date): JP 2001348168 011114
DESIGNATED STATES: AT; BE; CH; CY; DE; DK; ES; FI; FR; GB; GR; IE; IT; LI; LU; MC; NL; PT; SE; TR
EXTENDED DESIGNATED STATES: AL; LT; LV; MK; RO; SI
INTERNATIONAL PATENT CLASS: G06F-017/30

ABSTRACT EP 1313034 A2
A storage system (10) for providing a preferable data access performance by performing controls considering database management system (DBMS) execution information (38) or database process priorities, by acquiring static configurational information of a DBMS (110a, 110b) by means of a DBMS information acquisition and communication program (118),

a DBMS information communication section (116), and a host information setting program (130); acquiring DBMS execution information (38) by means of a query plan acquisition program (120), the DBMS information communication section (116), and a process performance management program (132); acquiring information on priorities of database processes given by the process performance management program (132); and storing them in disk I/O management information with process priorities (36, 36b), DBMS execution information (38), and DBMS data information (40), in which cache control section (54) in a storage system control program (50) controls a data cache (28) by referring the above information.

ABSTRACT WORD COUNT: 148

NOTE:

Figure number on first page: 1

LEGAL STATUS (Type, Pub Date, Kind, Text):

Application: 030521 A2 Published application without search report

Examination: 030521 A2 Date of request for examination: 20020327

LANGUAGE (Publication,Procedural,Application): English; English; English

FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	200321	802
SPEC A	(English)	200321	24306
Total word count - document A			25108
Total word count - document B			0
Total word count - documents A + B			25108

...SPECIFICATION destination. At this time, even if the access destination does not completely match the obtained **prefetched** area, it is determined to be included if a **distance** between them is equal to or smaller than a fixed number of blocks. This **distance** is **determined** by means of a difference in the data structure block ID number 716 and the ...

11/5,K/2 (Item 2 from file: 348)

DIALOG(R)File 348:EUROPEAN PATENTS

01568795

Cache memory operation

Cache-Speicherbetrieb

Fonctionnement d'une antememoire

PATENT ASSIGNEE:

STMicroelectronics Limited, (1828183), 1000 Aztec West, Almondsbury, Bristol BS32 4SQ, (GB), (Applicant designated States: all)

INVENTOR:

Southwell, Trefor, 12 Cossham ST, Bristol BS16 9EN, (GB)

Hedinger, Peter, Flat 3, 17 Regent Street, Clifton, Bristol BS8 4HW, (GB)

LEGAL REPRESENTATIVE:

Driver, Virginia Rozanne et al (58902), Page White & Farrer 54 Doughty Street, London WC1N 2LS, (GB)

PATENT (CC, No, Kind, Date): EP 1304619 A1 030423 (Basic)

APPLICATION (CC, No, Date): EP 2001308961 011022;

DESIGNATED STATES: DE; FR; GB; IT

EXTENDED DESIGNATED STATES: AL; LT; LV; MK; RO; SI

INTERNATIONAL PATENT CLASS: G06F-012/08

ABSTRACT EP 1304619 A1

A cache memory comprises a fetch engine arranged to issue fetch requests for accessing data items from locations in a main memory identified by access addresses in a program being executed, a pre-fetch

engine controlled to issue pre-fetch requests for speculatively accessing pre-fetch data items from locations in said main memory identified by addresses which are determined as being a number of locations from respective ones of said access addresses, and a calibrator arranged to selectively vary said number of locations.
ABSTRACT WORD COUNT: 82
NOTE:
Figure number on first page: 2

LEGAL STATUS (Type, Pub Date, Kind, Text):
Application: 030423 A1 Published application with search report
Examination: 031210 A1 Date of request for examination: 20031010
LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	200317	1155
SPEC A	(English)	200317	7483
Total word count - document A			8638
Total word count - document B			0
Total word count - documents A + B			8638

...SPECIFICATION Thus it can be understood that the pre-fetch calibration keeps the number of discarded **pre - fetches** to a minimum by optimising the **pre - fetch distance** . Furthermore, it ensures that a large percentage of the **pre - fetched** lines are actually used, which helps to reduce the number of cache conflicts. Thirdly it removes the requirement for the software to **determine** the **pre - fetch distance** . This allows standard applications and benchmarks to benefit from **pre - fetching** without requiring any software support. This means that there is no need to modify the...one write buffer. The pre-fetch back-off could always retain a different number of **pre - fetches** than one. In the **calculation** of **pre - fetch distance** 'd', the value of the issued **pre - fetch** counter could be divided by a different power of 2 or a power of a...

11/5,K/3 (Item 3 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

01108928
DATA PROCESSING CIRCUIT WITH CACHE MEMORY
DATENVERARBEITUNGS-SCHALTUNG MIT PUFFERSPEICHER
CIRCUIT DE TRAITEMENT DE DONNEES DOTE D'UNE ANTEMEMOIRE
PATENT ASSIGNEE:
Koninklijke Philips Electronics N.V., (200769), Groenewoudseweg 1, 5621 BA Eindhoven, (NL), (Proprietor designated states: all)
INVENTOR:
VAN DER WOLF, Pieter, Prof. Holstlaan 6, NL-5656 AA Eindhoven, (NL)
STRUIK, Pieter, Prof. Holstlaan 6, NL-5656 AA Eindhoven, (NL)
LEGAL REPRESENTATIVE:
de Jong, Durk Jan et al (84461), Internationaal Octrooibureau B.V., Prof. Holstlaan 6, 5656 AA Eindhoven, (NL)
PATENT (CC, No, Kind, Date): EP 996891 A2 000503 (Basic)
EP 996891 B1 030319
WO 99059070 991118
APPLICATION (CC, No, Date): EP 99914717 990429; WO 99IB785 990429
PRIORITY (CC, No, Date): EP 98201513 980508
DESIGNATED STATES: DE; FR; GB; NL
INTERNATIONAL PATENT CLASS: G06F-012/08
CITED PATENTS (EP B): WO -92/20027 A; US 5649144 A; US 5870599 A

NOTE:
No A-document published by EPO
LEGAL STATUS (Type, Pub Date, Kind, Text):

Application:	000503 A2	Published application without search report
Application:	20000112 A2	International application. (Art. 158(1))
Oppn None:	040310 B1	No opposition filed: 20031222
Grant:	030319 B1	Granted patent
Examination:	020102 A2	Date of dispatch of the first examination report: 20011109
Examination:	000503 A2	Date of request for examination: 20000208
Change:	010530 A2	Legal representative(s) changed 20010410
Change:	020703 A2	Title of invention (German) changed: 20020516
Lapse:	031008 B1	Date of lapse of European Patent in a contracting state (Country, date): NL 20030319,
Application:	20000112 A2	International application entering European phase

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	200312	478
CLAIMS B	(German)	200312	450
CLAIMS B	(French)	200312	506
SPEC B	(English)	200312	4418
Total word count - document A			0
Total word count - document B			5852
Total word count - documents A + B			5852

...SPECIFICATION may be needed. From this, an instruction for the cache management unit 16 can be **generated** for indicating a **distance** from the most recently accessed address over which **prefetched** numbers from addresses of the stream must be retained.
In one embodiment of the invention...

11/5,K/4 (Item 4 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00798925
Compiler for increased data cache efficiency
Kompilierer zur Verbesserung der Leistung von Datencachespeichern
Compilateur pour ameliorer l'efficacite d'une antememoire de donnees
PATENT ASSIGNEE:
Hewlett-Packard Company, A Delaware Corporation, (3016020), 3000 Hanover Street, Palo Alto, CA 94304, (US), (Proprietor designated states: all)
INVENTOR:
Santhanam, Vatsa, 2075 Anthony Dr., Campbell, CA 95008-2616, (US)
LEGAL REPRESENTATIVE:
Jehan, Robert et al (72663), Williams, Powell & Associates, 4 St Paul's Churchyard, London EC4M 8AY, (GB)
PATENT (CC, No, Kind, Date): EP 743598 A2 961120 (Basic)
EP 743598 A3 970723
EP 743598 B1 010926
APPLICATION (CC, No, Date): EP 96303454 960515;
PRIORITY (CC, No, Date): US 443653 950518
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: G06F-009/45
CITED REFERENCES (EP B):
COMPUTER ARCHITECTURE NEWS, vol. 22, no. 2, 1 April 1994, pages 223-232, XP000450353 CHEN T -F ET AL: "A PERFORMANCE STUDY OF SOFTWARE AND

HARDWARE DATA PREFETCHING SCHEMES"
PROCEEDINGS OF THE ANNUAL INTERNATIONAL SYMPOSIUM ON MICROARCHITECT, AUSTIN, DEC. 1 - 3, 1993, no. SYMP. 26, 1 December 1993, INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, pages 139-152, XP000447496
ABRAHAM S G ET AL: "PREDICTABILITY OF LOAD/STORE INSTRUCTION LATENCIES"
PROCEEDINGS OF THE REGION 10 ANNUAL INTERNATIONAL CONFERENCE (TENCO, SINGAPORE, 22 - 26 AUG., 1994, vol. VOL. 2, no. CONF. 9, 22 August 1994, CHAN T K Y (ED), pages 274-278, XP000529485 CHI C -H ET AL: "COMPILER DRIVEN DATA CACHE PREFETCHING FOR HIGH PERFORMANCE COMPUTERS"
ACM SIGPLAN NOTICES, vol. 26, no. 4, 8 April 1991, pages 40-52, XP000577434 CALLAHAN D ET AL: "SOFT WARE PREFETCHING"
ACM SIGPLAN NOTICES, vol. 27, no. 9, 1 September 1992, pages 62-73, XP000330590 MOWRY T C ET AL: "DESIGN AND EVALUATION OF A COMPILER ALGORITHM FOR PREFETCHING";

ABSTRACT EP 743598 A2
A compiler that facilitates efficient insertion of explicit data prefetch instructions into loop structures within applications uses simple address expression analysis to determine data prefetching requirements. Analysis and explicit data cache prefetch instruction insertion are performed by the compiler in a machine-instruction level optimizer to provide access to more accurate expected loop iteration latency information. Such prefetch instruction insertion strategy tolerates worst-case alignment of user data structures relative to data cache lines. Execution profiles from previous runs of an application are exploited in the insertion of prefetch instructions into loops with internal control flow. Cache line reuse patterns across loop iterations are recognized to eliminate unnecessary prefetch instructions. The prefetch insertion algorithm is integrated with other low-level optimization phases, such as loop unrolling, register reassociation, and instruction scheduling. An alternative embodiment of the compiler limits the insertion of explicit prefetch instructions to those situations where the lower bound on the achievable loop iteration latency is unlikely to be increased as a result of the insertion. (see image in original document)
ABSTRACT WORD COUNT: 193
NOTE:
Figure number on first page: 8

LEGAL STATUS (Type, Pub Date, Kind, Text):

Examination:	000510	A2	Date of dispatch of the first examination report: 20000323
Application:	961120	A2	Published application (A1with Search Report ;A2without Search Report)
Oppn None:	020918	B1	No opposition filed: 20020627
Assignee:	010328	A2	Transfer of rights to new applicant: Hewlett-Packard Company, A Delaware Corporation (3016020) 3000 Hanover Street Palo Alto, CA 94304 US
Change:	001122	A2	Legal representative(s) changed 20001005
Grant:	010926	B1	Granted patent
Search Report:	970723	A3	Separate publication of the European or International search report
Examination:	980325	A2	Date of filing of request for examination: 980123

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPAB96	865
CLAIMS B	(English)	200139	865
CLAIMS B	(German)	200139	811

CLAIMS B (French) 200139 1104
SPEC A (English) EPAB96 10179
SPEC B (English) 200139 10244
Total word count - document A 11045
Total word count - document B 13024
Total word count - documents A + B 24069

...SPECIFICATION vertical bar) B(i+1).eq(underscore)offset - B(i).eq(underscore)offset (vertical bar) <= **prefetch** memory **distance**

where the **prefetch** memory **distance** is **computed** as the product of PFID and the effective memory stride, M' for the equivalence class...

...of the cluster is 64-bytes (i.e. &A(i+7) - &A(i)). If the **prefetch** memory **distance** was **comp**uted earlier to be 128-bytes, i.e corresponding to a **prefetch** iteration **distance** of two, it is only necessary to insert three **prefetch** instructions to account for the entire span of this 8-member cluster. These three prefetches... candidates for small-stride equivalence classes. Basically, if two adjacent clusters are less than the **prefetch** memory **distance** apart, as **measured** between the trailing cluster's leader and the leading cluster's trailer, then the trailing...

...SPECIFICATION consideration iff

I B(i+1).eq(underscore)offset - B(i).eq(underscore)offset I <=== **prefetch** memory **distance** where the **prefetch** memory **distance** is **computed** as the product of PFID and the effective memory stride, M' for the equivalence class...of the cluster is 64-bytes (i.e. &A(i+7) - &A(i)). If the **prefetch** memory **distance** was **computed** earlier to be 128-bytes, i.e. corresponding to a **prefetch** iteration **distance** of two, it is only necessary to insert three **prefetch** instructions to account for the entire span of this 8-member cluster. These three prefetches...

...candidates for small-stride equivalence classes. Basically, if two adjacent clusters are less than the **prefetch** memory **distance** apart, as **measured** between the trailing cluster's leader and the leading cluster's trailer, then the trailing...

...CLAIMS 14. The method of any of Claims 8 to 13, further comprising the step of:

determining **prefetch** memory **distance** for the memory references associated with each equivalence class as the product of effective memory stride and **prefetch** iteration **distance** for the loop.

15. The method of any of Claims 8 to 14, further comprising...

...CLAIMS 11. The method of any of Claims 5 to 10, further comprising the step of:

determining **prefetch** memory **distance** for the memory references associated with each equivalence class as the product of effective memory stride and **prefetch** iteration **distance** for the loop.

12. The method of any of Claims 5 to 11, further comprising...

11/5,K/5 (Item 5 from file: 349)

01106222

POST-PASS BINARY ADAPTATION FOR SOFTWARE-BASED SPECULATIVE PRECOMPUTATION
ADAPTATION BINAIRE POST-PASSAGE DE CALCUL PRELIMINAIRE SPECULATIF PAR LOGICIEL

Patent Applicant/Assignee:
INTEL CORPORATION, 2200 Mission College Boulevard, Santa Clara, CA 95052, US, US (Residence), US (Nationality)
Inventor(s):
LIAO Steve Shih-Wei, 2671 Cowper Street, Palo Alto, CA 94306, US,
WANG Perry, 2907 Kiperash Drive, San Jose, CA 95133, US,
WANG Hong, 39877 Sundale Drive #105, Fremont, CA 94538, US,
HOFLEHNER Geolf, 2250 Monroe Street, No. 206, Santa Clara, CA 95050, US,
LAVERY Daniel, 3131 Homestead Road, No. 23A, Santa Clara, CA 95051, US,
SHEN John, 2303 Quail Bluff Place, San Jose, CA 95121, US,
Legal Representative:
MALLIE Michael J (et al) (agent), Blakely, Sokoloff, Tayor & Zafman, 7th Floor, 12400 Wilshire Boulevard, Los Angeles, CA 90025, US,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200427605 A2 20040401 (WO 0427605)
Application: WO 2003US27787 20030905 (PCT/WO US03027787)
Priority Application: US 2002245548 20020917
Designated States: AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NI NO NZ OM PG PH PL PT RO RU SC SD SE SG SK SL SY TJ TM TN TR TT TZ UA UG UZ VC VN YU ZA ZM ZW
(EP) AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LU MC NL PT RO SE SI SK TR
(OA) BF BJ CF CG CI CM GA GN GQ GW ML MR NE SN TD TG
(AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZM ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Main International Patent Class: G06F-009/40
Publication Language: English
Filing Language: English
Fulltext Availability:
Detailed Description
Claims
Fulltext Word Count: 10037

English Abstract
The latencies associated with cache misses or other long-latency instructions in a main thread are decreased through the use of a simultaneous helper thread. The helper thread is a speculative prefetch thread to perform a memory prefetch for the main thread. The instructions for the helper thread are dynamically incorporated into the main thread binary during post-pass operation of a compiler.

French Abstract
L'invention concerne les temps d'attente associes a l'absence d'informations dans l'antememoire ou d'autres instructions avec un temps d'attente long dans une unite d'execution principale, reduits par utilisation simultanee d'une unite d'execution auxiliaire. L'unite d'execution auxiliaire est une unite d'execution de prelecture speculative permettant d'executer une prelecture de memoire pour l'unite d'execution principale. Les instructions destines a l'unite d'execution auxiliaire sont incorporees de maniere dynamique dans la partie binaire de l'unite d'execution principale au cours de l'operation post-passage d'un compilateur.

Legal Status (Type, Date, Text)
Publication 20040401 A2 Without international search report and to be republished upon receipt of that report.

Fulltext Availability:
Detailed Description

Detailed Description
... while a slice that contains too much slack may result in early eviction of the **prefetched** data. Slack is the execution **distance** , **measured** , e.g., in terms of machine cycles, between execution of the prefetch instruction in the...

11/5,K/6 (Item 6 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

01011201 **Image available**
METHOD AND APPARATUS FOR SELECTING REFERENCES FOR PREFETCHING IN AN OPTIMIZING COMPILER
PROCEDE ET APPAREIL DE SELECTION DE REFERENCES POUR LIRE PAR ANTICIPATION UN COMPILATEUR D'OPTIMISATION
Patent Applicant/Assignee:
SUN MICROSYSTEMS INC, 4150 Network Circle, Santa Clara, CA 95054, US, US (Residence), US (Nationality)
Inventor(s):
TIRUMALAI Partha, 33916 Rowland Dr., Fremont, CA 94555, US,
KALOGEROPULOS Spiros, 258 East Main St. #C, Los Gatos, CA 95030, US,
RAJAGOPALAN Mahadevan, 311 Grau Dr., Fremont, CA 94536, US,
SONG Yonghong, 650 S. 8th St. #12, San Jose, CA 95112, US,
RAO Vikram, 62 Laurie Meadows Dr. Apt. 2, San Mateo, CA 94403, US,
Legal Representative:
PARK Richard (agent), 508 Second St., Ste. 201, Davis, CA 95616, US,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200341328 A2-A3 20030515 (WO 0341328)
Application: WO 2002US34144 20021024 (PCT/WO US0234144)
Priority Application: US 200152997 20011102
Designated States: JP
(EP) AT BE BG CH CY CZ DE DK EE ES FI FR GB GR IE IT LU MC NL PT SE SK TR
Main International Patent Class: G06F-009/00
International Patent Class: G06F-009/38; G06F-012/08
Publication Language: English
Filing Language: English
Fulltext Availability:
Detailed Description
Claims
Fulltext Word Count: 7014

English Abstract
One embodiment of the present invention provides a system that generates code to perform anticipatory prefetching for data references. During operation, the system receives code to be executed on a computer system. Next, the system analyzes the code to identify data references to be prefetched. This analysis can involve: using a two-phase marking process in which blocks that are certain to execute are considered before other blocks; and analyzing complex array subscripts. Next, the system inserts prefetch instructions into the code in advance of the identified data references. This insertion can involve: dealing with non-constant or unknown stride values; moving prefetch instructions into preceding basic

blocks; and issuing multiple prefetches for the same data reference.

French Abstract

La presente invention porte sur un systeme qui genere un code pour effectuer une prelecture des references de donnees. En fonctionnement, le systeme recoit un code a executer sur un systeme informatique. Puis, il analyse le code pour identifier des references de donnees a lire par anticipation. Cette analyse consiste a utiliser un procede de marquage en deux phases dans lequel des blocs qui sont surs d'etre executes sont pris en compte avant d'autres blocs ; et analyser des indices de reseaux complexes. Le systeme insere ensuite des instructions de prelecture dans le code precedant les references de donnees identifiees. Cette insertion peut consister a traiter des valeurs de cadence non constantes ou inconnues ; introduire des instructions de prelecture dans des blocs de base precedents et emettre des prelectures multiples pour la meme reference de donnees.

Legal Status (Type, Date, Text)

Publication	20030515	A2 Without international search report and to be republished upon receipt of that report.
Examination	20030619	Request for preliminary examination prior to end of 19th month from priority date
Search Rpt	20031113	Late publication of international search report
Republication	20031113	A3 With international search report.
Republication	20031113	A3 Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.
Republication	20031113	A3 Upon request of the applicant under Article 64(3)(c)(i).

Fulltext Availability:
Detailed Description

Detailed Description

... the above variation, while inserting the prefetch instruction based on the stride value, the system **calculates** a **prefetch** cover **distance** by dividing a cache line size by the stride value. The system also **calculates** a **prefetch** ahead **distance** as a function of a **prefetch** latency, the **prefetch** cover **distance** and an execution time of a loop. The system finally calculates a **prefetch** address by multiplying the stride value by the **prefetch** cover **distance** and the **prefetch** ahead ldistance and adding the result to an address accessed by the **prefetch** candidate.

One embodiment of the present invention provides a system for generating code to perform...stride function (step 608).

Next, the system performs a number of operations before calculating the **prefetch** address. In doing so, the system calculates a " **prefetch** ahead **distance** ", which indicates 1 5 how many loop iterations ahead to **prefetch** for. The system also considers the prefetch latency and the execution time of the loop.

The system starts by **calculating** the " **prefetch** cover **distance** " over the list of **prefetch** candidates (step 612). This **prefetch** cover **distance** is **calculated** as the cache line size divided by the stride. The **prefetch** cover **distance** indicates how many strides are covered by a single **prefetch** and is used to avoid prefetching the same cache line.

The system also computes the...

...the execution probability of the basic block relative to the loop header.

The system then **computes** the **prefetch** ahead **distance** using the following
function (step 614),
prefetch ahead **distance**
min(max(1, **prefetch** latency/(**prefetch** cover **distance** *etj)),
max(1, outstanding **prefetches** /number of **prefetch** streams)
Note that the "outstanding prefetches" value is a processor characteristic.

The system then calculates...

11/5,K/7 (Item 7 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00921118
SYSTEM AND METHOD FOR PREFETCHING DATA INTO A CACHE BASED ON MISS DISTANCE
SYSTEME ET PROCEDE POUR FAIRE UNE LECTURE ANTICIPEE DE DONNEES DANS UNE ANTEMEMOIRE SUR LA BASE D'UNE DISTANCE ENTRE DES ERREURS DE LECTURE
Patent Applicant/Assignee:
INTEL CORPORATION, 2200 Mission College Boulevard, Santa Clara, CA 95052, US, US (Residence), US (Nationality)
Inventor(s):
ABDALLAH Mohammad, 104 Austin Drive, Folsom, CA 95630, US,
AL-DAJANI Khalid, 3212 Sbaru Court, Sacramento, CA 95826, US,
Legal Representative:
CORSELLO Kenneth R (et al) (agent), Kenyon & Kenyon, Suite 600, 333 West San Carlos Street, San Jose, CA 95110, US,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200254230 A2-A3 20020711 (WO 0254230)
Application: WO 2001US49776 20011218 (PCT/WO US0149776)
Priority Application: US 2000749936 20001229
Designated States: AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ PH PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
(OA) BF BJ CF CG CI CM GA GN GQ GW ML MR NE SN TD TG
(AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZM ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Main International Patent Class: G06F-009/38
Publication Language: English
Filing Language: English
Fulltext Availability:
Detailed Description
Claims
Fulltext Word Count: 7521

English Abstract
A prefetcher to prefetch data for an instruction based on the distance between cache misses caused by the instruction. In an embodiment, the prefetcher includes a memory to store a **prefetch** table that contains one or more entries that include the **distance** between cache misses caused by an instruction. In a further embodiment, the addresss of data element sprefetched are **determined** based on the **distance** betweeen

cache misses recorded in the **prefetch** table for the instruction.

French Abstract

L'invention concerne un lecteur en amont destine a la lecture anticipee des donnees d'une instruction basee sur la distance existant entre les erreurs de lecture en antememoire causees par ladite instruction. Dans un mode de realisation, ce lecteur en amont comprend une memoire pour conserver une table de lecture anticipee qui comporte une ou plusieurs entrees contenant les erreurs de lecture en antememoire causees par l'instruction. Dans un autre mode de realisation, les adresses des elements de donnees prelus sont determinees sur la base de la distance entre les erreurs de lecture en antememoire enregistrees dans la table de lecture anticipee pour ladite instruction.

Legal Status (Type, Date, Text)

Publication	20020711	A2 Without international search report and to be republished upon receipt of that report.
Examination	20030213	Request for preliminary examination prior to end of 19th month from priority date
Correction	20030306	Corrections of entry in Section 1: under (30) replace "09/740,803" by "09/749,936"
Republication	20030306	A2 Without international search report and to be republished upon receipt of that report.
Correction	20030306	Corrections of entry in Section 1:
Search Rpt	20031211	Late publication of international search report
Republication	20031211	A3 With international search report.

Fulltext Availability:

Detailed Description

Claims

English Abstract

...caused by the instruction. In an embodiment, the prefetcher includes a memory to store a **prefetch** table that contains one or more entries that include the **distance** between cache misses caused by an instruction. In a further embodiment, the addresss of data element sprefetched are **determined** based on the **distance** betweeen cache misses recorded in the **prefetch** table for the instruction.

Detailed Description

... the table. In this embodiment, when a second miss later occurs for this instruction, the **prefetch** manager 122 will **determine** that the miss **distance** is equal to 30 (1030 - 1000) and will store this value in the miss distance...processor.

Embodiments of the present invention relate to a critical miss prefetcher. The critical miss **prefetcher** **prefetches** data for an instruction based on the **distance** between cache misses for the instruction.

The invention uses the predictability of the miss **distance** to **determine** the addresses to be **prefetched** .

Several embodiments of the present invention are specifically illustrated and/or described herein. However, it...

...addition, the invention may be used with physical address or linear addresses. In addition, the **prefetcher** may use a multiple of the miss **distance** when **determining** the address to be **prefetched** .

1 1

Claim
... two cache misses.

4 The prefetcher of claim 2, wherein the addresses of data elements **prefetched** are **determined** based on the **distance** between cache misses recorded in the **prefetch** table for the instruction.
5 The prefetcher of claim 2, wherein the prefetcher has a...

11/5,K/8 (Item 8 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00527718 **Image available**
DATA PROCESSING CIRCUIT WITH CACHE MEMORY
CIRCUIT DE TRAITEMENT DE DONNEES DOTE D'UNE ANTEMEMOIRE
Patent Applicant/Assignee:
KONINKLIJKE PHILIPS ELECTRONICS N V,
PHILIPS AB,
Inventor(s):
VAN DER WOLF Pieter,
STRUIK Pieter,
Patent and Priority Information (Country, Number, Date):
Patent: WO 9959070 A2 19991118
Application: WO 99IB785 19990429 (PCT/WO IB9900785)
Priority Application: EP 98201513 19980508
Designated States: JP AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE
Main International Patent Class: G06F-012/08
Publication Language: English
Fulltext Availability:
Detailed Description
Claims
Fulltext Word Count: 5093
English Abstract
The processing circuit contains a cache management unit which keeps information about a stream of addresses among addresses accessed by the processor. The cache management unit updates a current address for the stream in response to progress of execution of the program. The cache management unit makes selected storage locations in the cache memory available for reuse, a storage location in the cache memory which is in use for the data corresponding to the particular address being made available for reuse dependent on a position of the particular address relative to the current address.

French Abstract
Le circuit de traitement de l'invention est equipe d'une unite de gestion d'antememoire qui conserve des donnees relatives a un flux d'adresses parmi des adresses auxquelles le processeur a accede. Cette unite de gestion d'antememoire met a jour une adresse courante pour le flux, selon la progression du programme en cours d'execution, et libere des emplacements de stockage choisis dans l'antememoire en vue de leur reutilisation. Un emplacement de stockage dans l'antememoire actuellement utilise pour les donnees correspondant a l'adresse particuliere est immediatement liberee pour etre reutilise selon une position de ladite adresse particuliere relativement a l'adresse courante.
Fulltext Availability:
Detailed Description

Detailed Description

... may be needed. From this, an instruction for the cache management unit 16 can be **generated** for indicating a **distance** from the most recently accessed address over which **prefetched** numbers from addresses of the stream must be retained.

In one embodiment of the invention...
?

File 347:JAPIO Nov 1976-2004/Feb(Updated 040607)
(c) 2004 JPO & JAPIO
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200437
(c) 2004 Thomson Derwent

Set	Items	Description
S1	2238	PREFETCH? OR PRE()FETCH?
S2	532465	DISTANCE? ?
S3	56485	S2(3N)(DETERMIN? OR DET? ? OR CALCULAT? OR COMPUT??? ? OR COMPUTAT? OR MEASUR???????? ? OR GAUG??? ? ? OR ASSESS? OR APPRAIS?)
S4	5004	S2(3N)(ANALYS? OR ANALYZ? OR ANALYT? OR QUANTIF? OR QUANTITAT? OR DERIV? OR GENERAT???? ? OR MENSUR? OR EVALUAT? OR QUANTIS? OR QUANTIZ?)
S5	90859	PIPELIN? OR PIPE()LIN???? ?
S6	56965	TRAVERS???? ?
S7	2	S1(2N)S2
S8	8	S1(10N)S2
S9	4	S8 AND S3:S4
S10	4	IDPAT (sorted in duplicate/non-duplicate order)
S11	4	IDPAT (primary/non-duplicate records only)
S12	28364	LIBRARY? OR LIBRARIES
S13	469875	OBJECT? ? OR OBJECTORIENT? OR OO OR OOP? ? OR OOP?? ? OR OOF OR OOPLA OR OOPL OR ODBM? ? OR OOAD
S14	167	OODB OR ODMS OR OOL OR OOOS OOPS OR OOT OR OOS
S15	422	LINKED()LIST? ? OR SUBLIST?
S16	499347	TREE OR TREES OR FOREST? ? OR SUBTREE? ? OR CHILDREN OR CHILD OR NODE OR NODES OR SUBNODE? OR TREEMAP? OR LEAVES OR LEAF? ? OR BRANCH?
S17	56273	SUBBRANCH? OR MULTIBRANCH? OR PARENT? ? OR OFFSPRING? OR OFF()SPRING? ? OR ANCEST?R? ? OR DESCEND?NT? ?
S18	55219	PYRAMID? OR CASCAD??? ? OR HIERARCH?
S19	339	S1(7N)S12:S18
S20	1021	S6(7N)S12:S18
S21	1	S19 AND S20
S22	30	S19:S20(20N)S5
S23	27	S22 AND S1
S24	28	(S21 OR S23) NOT S11
S25	28	IDPAT (sorted in duplicate/non-duplicate order)
S26	28	IDPAT (primary/non-duplicate records only)

26/9/17 (Item 17 from file: 347)
DIALOG(R)File 347:JAPIO

05014190 **Image available**
LOOP OPTIMIZATION SYSTEM

PUB. NO.: 07-306790 [JP 7306790 A]
PUBLISHED: November 21, 1995 (19951121)
INVENTOR(s): KAWASAKI TORU
APPLICANT(s): HITACHI LTD [000510] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 06-100466 [JP 94100466]
FILED: May 16, 1994 (19940516)
INTL CLASS: [6] G06F-009/45; G06F-009/38; G06F-012/08
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units); 45.2 (INFORMATION PROCESSING -- Memory Units)

ABSTRACT

PURPOSE: To facilitate application by making a compiler automatically judge whether optimization by the issue of a **prefetch** load instruction effective to the dynamic reduction of a cache miss penalty and perform the optimization.

CONSTITUTION: Through an analytic path (105) for software pipelining optimization, information the number and data type of objects and the number of necessary registers which are required to apply **prefetch** loading is gathered, After a loop to be pipelined is selected (106), a developing method is determined by **prefetch** loading execution selection in 107 so that large-area register allocation (109) is not impeded; and an expansion number is divided by the number of objects so as to equalize the distance between preloading and preloading, thereby dividing parts as many as the **objects** . Then an intermediate word for **prefetch** loading indication is interposed right before each of divided part at the time of intermediate code output (108) after **pipelining** conversion.
?

? t11/9/all

11/9/1 (Item 1 from file: 350)
DIALOG(R)File 350:Derwent WPIX

014727424 **Image available**
WPI Acc No: 2002-548128/200258
Related WPI Acc No: 2002-500674; 2002-655655
XRPX Acc No: N02-433927
Prefetching data into a cache based on miss distance **e.g. for** computer **processors, where** prefetcher **to** prefetch **data for an instruction is based on the** distance **between cache misses caused by the instruction**
Patent Assignee: INTEL CORP (ITLC); ABDALLAH M A (ABDA-I); AL-DAJANI K (ALDA-I)
Inventor: ABDALLAH M; AL-DAJANI K; ABDALLAH M A
Number of Countries: 100 Number of Patents: 009
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
WO 200254230	A2	20020711	WO 2001US49776	A	20011218	200258 B
US 20020087800	A1	20020704	US 2000749936	A	20001229	200319
US 6584549	B2	20030624	US 2000749936	A	20001229	200343
EP 1346281	A2	20030924	EP 2001988368	A	20011218	200363
			WO 2001US49776	A	20011218	
US 20030196046	A1	20031016	US 2000749936	A	20001229	200369
			US 2003427908	A	20030502	
TW 541498	A	20030711	TW 2001131353	A	20011218	200406
US 6701414	B2	20040302	US 2000749936	A	20001229	200417
			US 2003427908	A	20030502	
AU 2002241682	A1	20020716	AU 2002241682	A	20011218	200427
CN 1484788	A	20040324	CN 2001821629	A	20011218	200437

Priority Applications (No Type Date): US 2000749936 A 20001229; US 2003427908 A 20030502
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 200254230	A2	E	26	G06F-009/38	
Designated States (National): AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ PH PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW					
Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW MZ NL OA PT SD SE SL SZ TR TZ UG ZM ZW					
US 20020087800	A1			G06F-012/08	
US 6584549	B2			G06F-012/00	
EP 1346281	A2	E		G06F-009/38	Based on patent WO 200254230
Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI TR					
US 20030196046	A1			G06F-012/00	Cont of application US 2000749936 Cont of patent US 6584549
TW 541498	A			G06F-009/38	
US 6701414	B2			G06F-012/00	Cont of application US 2000749936 Cont of patent US 6584549
AU 2002241682	A1			G06F-009/38	Based on patent WO 200254230
CN 1484788	A			G06F-009/38	

Abstract (Basic): WO 200254230 A2
NOVELTY - **Prefetcher** to **prefetch** data for an instruction is based on the **distance** between cache misses caused by the instruction, where the **prefetcher** includes a memory to store a **prefetch** table

that contains one or more entries that include the **distance** between cache misses caused by an instruction. The addresses of data elements **prefetched** are **determined** based on the **distance** between cache misses recorded in the **prefetch** table for the instruction.

DETAILED DESCRIPTION - INDEPENDENT CLAIM included for the following:processor; method of managing data prefetching; machine-readable medium; prefetch system

USE - For computer processors.

ADVANTAGE - Computer systems operate more efficiently as number of cache misses are minimized.

DESCRIPTION OF DRAWING(S) - The diagram shows a computer system having prefetcher

cache (130)

prefetcher (120)

pp; 26 DwgNo 1/8

Title Terms: DATA; CACHE; BASED; MISS; DISTANCE; COMPUTER; PROCESSOR; DATA; INSTRUCTION; BASED; DISTANCE; CACHE; MISS; CAUSE; INSTRUCTION

Derwent Class: T01

International Patent Class (Main): G06F-009/38; G06F-012/00; G06F-012/08

File Segment: EPI

Manual Codes (EPI/S-X): T01-F03A; T01-F03B; T01-H03A; T01-S03

11/9/2 (Item 2 from file: 350)

DIALOG(R)File 350:Derwent WPIX

014432624 **Image available**

WPI Acc No: 2002-253327/200230

Related WPI Acc No: 1995-368109; 1998-055414; 1999-443547

XRPX Acc No: N02-195436

Microprocessor for video graphics computer **systems, has pixel** distance computation **circuit and multiplier circuit which receive only one decoded instruction from prefetch dispatch unit in given clock cycle**

Patent Assignee: SUN MICROSYSTEMS INC (SUNM)

Inventor: KOHN L D; VAN HOOK T J; YUNG R

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 6279099	B1	20010821	US 94236572	A	19940429	200230	B
			US 96638390	A	19960426		
			US 99244187	A	19990204		
			US 2000648250	A	20000825		

Priority Applications (No Type Date): US 96638390 A 19960426; US 94236572 A 19940429; US 99244187 A 19990204; US 2000648250 A 20000825

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6279099	B1		35	G06F-015/00	CIP of application US 94236572 Div ex application US 96638390 Cont of application US 99244187 CIP of patent US 5734874

Abstract (Basic): US 6279099 B1

NOVELTY - An arithmetic logic unit (ALU) (26) and multiplier unit (28) are connected to a **prefetch** and dispatch unit (PDU) (46). A pixel **distance computation** circuit **calculates** difference between multiple pairs of pixels, and is parallely positioned to multiplier circuit so as to receive only one decoded instruction from the PDU in a given clock cycle.

USE - Microprocessor for video graphics computer systems in which three-dimensional images are displayed or obtained as print outs through printers or broadcasted on TV monitors.

ADVANTAGE - The pixel **distance computation** circuit and the multiplier circuit receive only one decoded instruction from the PDU in a given clock cycle due to which need for multiple pixel accesses and compare operations with subsequent writing into the destination is reduced. Also, number rate of memory accesses is reduced.

DESCRIPTION OF DRAWING(S) - The figure shows the microprocessor architecture of graphics computer system.

FP graphics ALU unit (26)

FP graphics multiple unit (28)

Prefetch and dispatch unit (46)

pp; 35 DwgNo 1/11

Title Terms: MICROPROCESSOR; VIDEO; GRAPHIC; COMPUTER; SYSTEM; PIXEL; DISTANCE; COMPUTATION; CIRCUIT; MULTIPLIER; CIRCUIT; RECEIVE; ONE; DECODE ; INSTRUCTION; DISPATCH; UNIT; CLOCK; CYCLE

Derwent Class: T01; W03

International Patent Class (Main): G06F-015/00

File Segment: EPI

Manual Codes (EPI/S-X): T01-C04; T01-C05A; T01-E02B; T01-E02D; T01-F03; T01-J10C4; T01-K; W03-A

11/9/3 (Item 3 from file: 350)

DIALOG(R)File 350:Derwent WPIX

011009374 **Image available**

WPI Acc No: 1996-506324/199650

XRPX Acc No: N96-426574

Instruction memory limit check method for microporcessors - involves maintaining value of distance **to end of segment and checking at** prefetch **that value is not reduced beyond zero**

Patent Assignee: NAT SEMICONDUCTOR CORP (NASC)

Inventor: DIVIVIER R; NEMIROVSKY M

Number of Countries: 018 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 9635165	A1	19961107	WO 96US6146	A	19960501	199650	B
EP 771442	A1	19970507	EP 96913319	A	19960501	199723	
			WO 96US6146	A	19960501		

Priority Applications (No Type Date): US 95445564 A 19950506

Cited Patents: 3.Jnl.Ref; EP 36093

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 9635165	A1	E	20	G06F-009/38	

Designated States (National): DE KP

Designated States (Regional): AT BE CH DE DK ES FI FR GB GR IE IT LU MC NL PT SE

EP 771442	A1	E	1	G06F-009/38	Based on patent WO 9635165

Designated States (Regional): DE FR GB

Abstract (Basic): WO 9635165 A

The microprocessor has a pipelined structure that includes a prefetch stage, decode stage and execute stage. A memory segmentation structure is also used and limits instruction space. The execute stage includes a **calculation** to form the **distance** SEGSPACE between the current instruction pointer and the end of the current segment.

The prefetch stage retreives instructions from the branch address

sequentially. As bytes are retreived the SEGSPACE value is decremented by the number of bytes retreived. If this reaches zero a potential violation is flagged. The voilation is acted on conditionally only after prior instructions are completed.

ADVANTAGE - Provides an efficient method for detecting and reporting instruction segment limit violations.

Dwg.3/4

Title Terms: INSTRUCTION; MEMORY; LIMIT; CHECK; METHOD; MAINTAIN; VALUE; DISTANCE; END; SEGMENT; CHECK; VALUE; REDUCE; ZERO
Derwent Class: T01
International Patent Class (Main): G06F-009/38
File Segment: EPI
Manual Codes (EPI/S-X): T01-F03B; T01-G02; T01-H01A

11/9/4 (Item 4 from file: 350)

007087811
WPI Acc No: 1987-087808/198713
XRPX Acc No: N87-065880

Magnetic tape system which command prefetch - has reel-to-reel tape drive and prefetch control including memory storing write commands from host system

Patent Assignee: FUJITSU LTD (FUIT)
Inventor: ISHIGURO M; OHWA N
Number of Countries: 008 Number of Patents: 008
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
EP 213703	A	19870311	EP 86305345	A	19860711	198713	B
AU 8660031	A	19870319				198718	
US 4757396	A	19880712	US 86883630	A	19860709	198830	
CA 1273710	A	19900904				199041	
CA 1280510	C	19910219				199113	
EP 213703	B1	19930915	EP 86305345	A	19860711	199337	
DE 3689021	G	19931021	DE 3689021	A	19860711	199343	
			EP 86305345	A	19860711		
KR 9204769	B1	19920615	KR 865635	A	19860712	199351	

Priority Applications (No Type Date): JP 85235807 A 19851022; JP 85153831 A 19850712; JP 85214248 A 19850927; JP 85222498 A 19851005
Cited Patents: 2.Jnl.Ref; A3...9002; DE 3232339; EP 59799; EP 96456; No-SR.Pub
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 213703	A	E	129		
Designated States (Regional): DE FR GB NL					
US 4757396	A		80		
EP 213703	B1	E	89	G06F-003/06	
Designated States (Regional): DE FR GB NL					
DE 3689021	G			G06F-003/06	Based on patent EP 213703
KR 9204769	B1			G06F-013/10	

Abstract (Basic): EP 213703 A

The write commands stored in a memory of a prefetch control include commands which are not accompanied by write data from the host system. A second memory stores write and read data from the magnetic tape. A motor drive is operated according to write commands received in advance from the host system and stored in the first memory by a drive controller.

The controller calculates the running time of the tape from the

diameters of tape wound on the reels and provides an alarm signal to warn of the imminence of the end of the tape. The prefetch control reduces the prefetching of write commands stored in the first memory after reception of an alarm signal warning of the near-approach of the end of the tape.

ADVANTAGE - Processing efficiency is enhanced by effecting command prefetching close to end of magnetic tape.

Abstract (Equivalent): EP 213703 B

A magnetic tape system comprising:- a drive means (1) having motors (10a, 10b) for driving a pair of reels (11, 12) on which is wound magnetic tape (16) having a beginning-of-tape (BOT) mark and end-of-tape (EOT) mark, and having a magnetic head (14); a drive control means (2), operatively connected to said drive means, for controlling said drive means based on given commands so as to write data on or read data from said magnetic tape; a prefetch control means (3), operatively connected to said drive control means and a host system (200) requesting access to the magnetic tape, and having a first memory (31b) which stores a plurality of write commands from said host system, including write commands which are not accompanied with write data from said host system, a second memory (34) which stores a plurality of write data, and read data from the magnetic tape, for operating said drive means based on write commands received in advance from said host system and stored in said first memory via said drive control means, writing data on or reading data from the magnetic tape biased on commands from the host system, and reporting results to the host system; characterised in that:- said drive control means is operable to **calculate** a running **distance** of the magnetic tape based on daimeters of the magnetic tape wound on said reels, and to issue an early-warning-area (EWA) signal indicating the approach of the end of the magnetic tape before detection of said EOT mark; and in that said prefetch control means (3) reduces prefetching of write commands stored in said first memory (31b) after reception of a first said EWA signal.

(Dwg.5/39

Abstract (Equivalent): US 4757396 A

The magnetic tape system has a drive control unit (2, 20, 21) for controlling the drive mechanism (1) based on commands to write data on the read data from the magnetic tape and a prefetch control unit. The prefetch control unit stores a number of write commands from a host system. The drive control unit operates the drive mechanism based on the write commands stored in the first memory.

The running **distance** of the magnetic tape may be calculated and **prefetch** control performed until the calculated value reaches a value corresponding to a second EWA position before the end of the magnetic tape. When the second EWA position is reached, an EOT signal is issued to the host system and prefetched write commands are no longer accepted.

ADVANTAGE - Enables execution of all prefetched write commands, reliable, enables efficient streaming operation in EWA

Title Terms: MAGNETIC; TAPE; SYSTEM; COMMAND; REEL-TO-REEL; TAPE; DRIVE; CONTROL; MEMORY; STORAGE; WRITING; COMMAND; HOST; SYSTEM
Derwent Class: P84; T01; T03
International Patent Class (Main): G06F-003/06; G06F-013/10
International Patent Class (Additional): G03F-003/06; G06F-000/01; G11B-015/18; G11B-027/30; G11S-005/09
File Segment: EPI; EngPI
Manual Codes (EPI/S-X): T01-C01; T03-J; T03-N04

```
File   9:Business & Industry(R) Jul/1994-2004/Jun 14
         (c) 2004  The Gale Group
File  16:Gale Group PROMT(R) 1990-2004/Jun 15
         (c) 2004 The Gale Group
File  47:Gale Group Magazine DB(TM) 1959-2004/Jun 14
         (c) 2004 The Gale group
File 148:Gale Group Trade & Industry DB 1976-2004/Jun 15
         (c)2004 The Gale Group
File 160:Gale Group PROMT(R) 1972-1989
         (c) 1999 The Gale Group
File 275:Gale Group Computer DB(TM) 1983-2004/Jun 15
         (c) 2004 The Gale Group
File 570:Gale Group MARS(R) 1984-2004/Jun 15
         (c) 2004 The Gale Group
File 621:Gale Group New Prod.Annou.(R) 1985-2004/Jun 15
         (c) 2004 The Gale Group
File 636:Gale Group Newsletter DB(TM) 1987-2004/Jun 14
         (c) 2004 The Gale Group
File 649:Gale Group Newswire ASAP(TM) 2004/Jun 14
         (c) 2004 The Gale Group

Set     Items   Description
S1       3509   PREFETCH? OR PRE()FETCH?
S2     799628   DISTANCE? ?
S3      21043   S2(3N)(DETERMIN? OR DET? ? OR CALCULAT? OR COMPUT??? ? OR -
             COMPUTAT? OR MEASUR???????? ? OR GAUG??? ? ? OR ASSESS? OR AP-
             PRAIS?)
S4       5083   S2(3N)(ANALYS? OR ANALYZ? OR ANALYT? OR QUANTIF? OR QUANTI-
             TAT? OR DERIV? OR GENERAT???? ? OR MENSUR? OR EVALUAT? OR QUA-
             NTIS? OR QUANTIZ?)
S5     617740   PIPELIN? OR PIPE()LIN???? ?
S6      53383   TRAVERS???? ?
S7          0   S1(2N)S2
S8          7   S1(10N)S2
S9          0   S8(S)S3:S4
S10    968802   LIBRARY? OR LIBRARIES
S11    563661   OBJECT? ? OR OBJECTORIENT? OR OO OR OOP? ? OR OOP?? ? OR O-
             OF OR OOPLA OR OOPL OR ODBM? ? OR OOAD
S12     13441   OODB OR ODMS OR OOL OR OOOS OR OOPS OR OOT OR OOS
S13      1479   LINKED()LIST? ? OR SUBLIST?
S14   2725510   TREE OR TREES OR FOREST? ? OR SUBTREE? ? OR CHILDREN OR CH-
             ILD OR NODE OR NODES OR SUBNODE? OR TREEMAP? OR LEAVES OR LEA-
             F? ?
S15   1730873   PARENT? ? OR OFFSPRING? OR OFF()SPRING? ? OR ANCEST?R? ? OR
              DESCEND?NT? ?
S16    274157   PYRAMID? OR CASCAD??? ? OR HIERARCH?
S17        45   S1(7N)S10:S16
S18      1486   S6(7N)S10:S16
S19         0   S17(S)S18
S20        31   S17:S18(20N)S5
S21         3   S20(S)S1
S22         2   RD (unique items)

 22/3,K/1     (Item 1 from file: 16)
DIALOG(R)File  16:Gale Group PROMT(R)
(c) 2004 The Gale Group. All rts. reserv.

06356425    Supplier Number: 54693101  (USE FORMAT 7 FOR FULLTEXT)
What's the cache?(Internet/Web/Online Service Information)
Computer Weekly, p58
May 13, 1999
```

Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 1161

... by cache suppliers to maximise performance. This is more efficient than generic file stores and **object** **prefetching** or **pipelining** , where all related **object** on a page are loaded into cache in one go
* Transparent caching combines Web caches...

22/3,K/2 (Item 1 from file: 148)
DIALOG(R)File 148:Gale Group Trade & Industry DB

10451652 SUPPLIER NUMBER: 21114725 (USE FORMAT 7 OR 9 FOR FULL TEXT)
RISC And CISC Processors Compete For Embedded Applications.
Bursky, Dave
Electronic Design, v46, n19, p83(1)
August 17, 1998
ISSN: 0013-4872 LANGUAGE: English RECORD TYPE: Fulltext
WORD COUNT: 4191 LINE COUNT: 00335

... includes a prefetch and dispatch unit that fetches instructions before they are needed in the **pipeline** . The instructions can be **prefetched** from all levels of the memory **hierarchy** . To allow **prefetches** across conditional branches, a branch-prediction scheme is implemented in hardware.
All the execution units...

```
File 696:DIALOG Telecom. Newsletters 1995-2004/Jun 14
         (c) 2004 The Dialog Corp.
File  15:ABI/Inform(R) 1971-2004/Jun 14
         (c) 2004 ProQuest Info&Learning
File  98:General Sci Abs/Full-Text 1984-2004/Jun
         (c) 2004 The HW Wilson Co.
File 484:Periodical Abs Plustext 1986-2004/Jun W1
         (c) 2004 ProQuest
File 813:PR Newswire 1987-1999/Apr 30
         (c) 1999 PR Newswire Association Inc
File 613:PR Newswire 1999-2004/Jun 15
         (c) 2004 PR Newswire Association Inc
File 635:Business Dateline(R) 1985-2004/Jun 15
         (c) 2004 ProQuest Info&Learning
File 810:Business Wire 1986-1999/Feb 28
         (c) 1999 Business Wire
File 610:Business Wire 1999-2004/Jun 15
         (c) 2004 Business Wire.
File 369:New Scientist 1994-2004/Jun W1
         (c) 2004 Reed Business Information Ltd.
File 370:Science 1996-1999/Jul W3
         (c) 1999 AAAS
File  20:Dialog Global Reporter 1997-2004/Jun 15
         (c) 2004 The Dialog Corp.
File 624:McGraw-Hill Publications 1985-2004/Jun 14
         (c) 2004 McGraw-Hill Co. Inc
File 634:San Jose Mercury  Jun 1985-2004/Jun 12
         (c) 2004 San Jose Mercury News
File 647:CMP  Computer Fulltext 1988-2004/Jun W1
         (c) 2004 CMP Media, LLC
File 674:Computer News Fulltext 1989-2004/Jun W2
         (c) 2004 IDG Communications

Set     Items   Description
S1       1214   PREFETCH? OR PRE()FETCH?
S2     828982   DISTANCE? ?
S3      19358   S2(3N)(DETERMIN? OR DET? ? OR CALCULAT? OR COMPUT??? ? OR -
             COMPUTAT? OR MEASUR???????? ? OR GAUG??? ? ? OR ASSESS? OR AP-
             PRAIS?)
S4       5385   S2(3N)(ANALYS? OR ANALYZ? OR ANALYT? OR QUANTIF? OR QUANTI-
             TAT? OR DERIV? OR GENERAT???? ? OR MENSUR? OR EVALUAT? OR QUA-
             NTIS? OR QUANTIZ?)
S5     528438   PIPELIN? OR PIPE()LIN???? ?
S6      57246   TRAVERS???? ?
S7          1   S1(2N)S2
S8          2   S1(10N)S2
S9          1   S8(S)S3:S4
S10         0   S9(S)S6
S11         0   S9(S)S5
S12    639390   LIBRARY? OR LIBRARIES
S13    521178   OBJECT? ? OR OBJECTORIENT? OR OO OR OOP? ? OR OOP?? ? OR O-
            OF OR OOPLA OR OOPL OR ODBM? ? OR OOAD
S14     16079   OODB OR ODMS OR OOL OR OOOS OR OOPS OR OOT OR OOS
S15       438   LINKED()LIST? ? OR SUBLIST?
S16   4205640   TREE OR TREES OR FOREST? ? OR SUBTREE? ? OR CHILDREN OR CH-
             ILD OR NODE OR NODES OR SUBNODE? OR TREEMAP? OR LEAVES OR LEA-
             F? ?
S17   1764533   PARENT? ? OR OFFSPRING? OR OFF()SPRING? ? OR ANCEST?R? ? OR
              DESCEND?NT? ?
S18    290221   PYRAMID? OR CASCAD??? ? OR HIERARCH?
S19        32   S1(7N)S12:S18
```

S20 1338 S6(7N)S12:S18
S21 0 S19(S)S20
S22 15 S19:S20(20N)S5
S23 2 S22(S)S1
S24 2 RD (unique items)
? t24/3,k/

24/3,K/1 (Item 1 from file: 813)
DIALOG(R)File 813:PR Newswire

0904414 LAM035
SILICON GRAPHICS UNVEILS NEW FAMILY OF CHALLENGE SERVERS FOR THE ENTERPRISE

DATE: January 22, 1996 12:30 EST WORD COUNT: 2,249

...memory access latency for real applications by employing optimizations for advanced superscalar architecture and memory **hierarchies** . Optimizations include software **pipelining** , automatic loop blocking, **prefetching** , extensive pointer aliasing and comprehensive support for interprocedural analysis of programs.

"The next generation compilers...
? t24/3,k/2

24/3,K/2 (Item 1 from file: 635)
DIALOG(R)File 635:Business Dateline(R)

0666555 96-23522
Silicon Graphics introduces powerful new technologies across entire product line
Henderson, Virginia
PR Newswire (New York, NY, US) s1 p1
PUBL DATE: 960122
WORD COUNT: 6,481
DATELINE: Mountain View, CA, US, Pacific

TEXT:

...memory access latency for real applications by employing optimizations for advanced superscalar architecture and memory **hierarchies** . Optimizations include software **pipelining** , automatic loop blocking, **prefetching** , extensive pointer aliasing and comprehensive support for interprocedural analysis of programs.

"The next generation compilers...
?

9/3,K/1 **(Item 1 from file: 484)**
DIALOG(R)File 484:Periodical Abs Plustext

04978447 SUPPLIER NUMBER: 68546547 (USE FORMAT 7 OR 9 FOR FULLTEXT)
Data prefetch mechanisms
Vanderwiel, Steven P; Lilja, David J
ACM Computing Surveys (ACI), v32 n2, p174-199, p.26
Jun 2000
ISSN: 0360-0300 JOURNAL CODE: ACI
DOCUMENT TYPE: Feature
LANGUAGE: English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 12304

TEXT:
... In the undirected mode, the next sequential line is prefetched. In the directed mode, the **prefetch** direction (forward or backward) and **distance** can be **determined** by the pre/post-increment amount encoded in the load or store instructions. That is...

File 256:SoftBase:Reviews,Companies&Prods. 82-2004/May
(c)2004 Info.Sources Inc

```
Set     Items   Description
S1         15   PREFETCH? OR PRE()FETCH?
S2       1373   DISTANCE? ?
S3         83   S2(3N)(DETERMIN? OR DET? ? OR CALCULAT? OR COMPUT??? ? OR -
                COMPUTAT? OR MEASUR???????? ? OR GAUG??? ? ? OR ASSESS? OR AP-
                PRAIS?)
S4         16   S2(3N)(ANALYS? OR ANALYZ? OR ANALYT? OR QUANTIF? OR QUANTI-
                TAT? OR DERIV? OR GENERAT???? ? OR MENSUR? OR EVALUAT? OR QUA-
                NTIS? OR QUANTIZ?)
S5        233   PIPELIN? OR PIPE()LIN???? ?
S6        197   TRAVERS???? ?
S7          0   S1(2N)S2
S8          0   S1(10N)S2
S9       5769   LIBRARY? OR LIBRARIES
S10     11644   OBJECT? ? OR OBJECTORIENT? OR OO OR OOP? ? OR OOP?? ? OR O-
             OF OR OOPLA OR OOPL OR ODBM? ? OR OOAD
S11         4   OODB OR ODMS OR OOL OR OOOS OOPS OR OOT OR OOS
S12         9   LINKED()LIST? ? OR SUBLIST?
S13      3714   TREE OR TREES OR FOREST? ? OR SUBTREE? ? OR CHILDREN OR CH-
             ILD OR NODE OR NODES OR SUBNODE? OR TREEMAP? OR LEAVES OR LEA-
             F? ?
S14       757   PARENT? ? OR OFFSPRING? OR OFF()SPRING? ? OR ANCEST?R? ? OR
              DESCEND?NT? ?
S15      1829   PYRAMID? OR CASCAD??? ? OR HIERARCH?
S16         1   S1(7N)S9:S15
S17        17   S6(7N)S9:S15
S18         0   S16 AND S17
S19         0   S16:S17(20N)S5
S20         0   S16:S17 AND S5
S21         1   S16:S17 AND S1
? t21/7
```

21/7/1

DIALOG(R)File 256:SoftBase:Reviews,Companies&Prods.
(c)2004 Info.Sources Inc. All rts. reserv.

01058408 DOCUMENT TYPE: Product

PRODUCT NAME: Enterprise DICOM Storage & Archive Manager (058408)

Emageon (707805)
1200 Corporate Dr #400
Birmingham, AL 35242 United States

RECORD TYPE: Directory

CONTACT: Sales Department

Enterprise DICOM Storage & Archive Manager from Emageon brings DICOM medical images and clinical information to any remote platform. The Enterprise DICOM software lets organizations extend existing PACS infrastructure and lower deployment costs. The superDICOMarchive component supports a distributed image cache that speeds TCP/IP image access, with images cached in an Active Patient Image Repository (APIR). The superDICOMarchive offers real-time configuration options and 24/7 uptime. Enterprise DICOM Archive Manager allows images to be accessed over intranets or through the Internet, employing authentication tools to protect data. The module has been benchmarked at 55 million DICOM images

and over 300 concurrent DICOM associations. Enterprise DICOM Storage & Archive Manager's Enterprise DICOM Distribution Manager is an open system that streamlines DICOM diagnostic imaging workflows. The Proxy component offers quick retrievals of objects across distributed environments. Router provides enterprisewide routing control. The **Prefetcher** component searches archives for related DICOM **objects** , and the Deletor allows the Proxy to delete cached studies. The Deletor references a configurable overflow threshold in deleting objects. The solution's other key module is the Enterprise Worklist Manager, which provides physicians with worklist storage, access, and creation options. Newer features include support for cardiology syntaxes, JPEG2000 DICOM, and HL7.

REVISION DATE: 20031015
?

```
File 347:JAPIO Nov 1976-2004/Feb(Updated 040607)
         (c) 2004 JPO & JAPIO
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200437
         (c) 2004  Thomson Derwent
File 348:EUROPEAN PATENTS 1978-2004/Jun W02
         (c) 2004 European Patent Office
File 349:PCT FULLTEXT 1979-2002/UB=20040610,UT=20040603
         (c) 2004 WIPO/Univentio

Set     Items   Description
S1          0   AU=COLDEWEY D?
```

	Hits	Search Text	DBs	Time Stamp ▵
1	7922	707/100-102,104.1.ccls.	USPAT; US-PGPUB	2004/08/13 14:27
2	2477	707/200,201.ccls.	USPAT; US-PGPUB	2004/08/13 14:27
3	6272	(creat$4 or build$4 or generat$4) with parallel and partition$2	USPAT	2004/08/13 14:42
4	6	determin$4 with prefetch adj2 distance$2	USPAT	2004/08/13 14:45
5	42	insert with instruction$2 with loop$2	USPAT	2004/08/13 14:46
6	11343	707/1-4,10.ccls.	USPAT; US-PGPUB	2004/08/13 14:50
7	2279	717/106-109,124-135.ccls.	USPAT; US-PGPUB	2004/08/13 14:50
8	2	(determin$4 with prefetch adj2 distance$2) and (insert with instruction$2 with loop$2)	USPAT	2004/08/13 14:55
9	17870	707/1-4,10.ccls. or 707/100-102,104.1.ccls. or 707/200,201.ccls.	USPAT; US-PGPUB	2004/08/13 14:55
10	2	(determin$4 with prefetch adj2 distance$2) and (insert with instruction$2 with loop$2)	USPAT; US-PGPUB	2004/08/13 14:55
11	37	(creat$4 or build$4 or generat$4) with parallel adj2 structure$2 and partition$2	USPAT	2004/08/13 14:56
12	63	pipelin$4 with travers$3 and partition$2	USPAT; US-PGPUB	2004/08/13 14:57
13	4	((707/1-4,10.ccls. or 707/100-102,104.1.ccls. or 707/200,201.ccls.) or ((determin$4 with prefetch adj2 distance$2) and (insert with instruction$2 with loop$2))) and (creat$4 or build$4 or generat$4) with parallel adj2 structure$2 and partition$2	USPAT; US-PGPUB	2004/08/13 14:59
14	13	pipelin$4 with travers$3 and partition$2 and tree$2	USPAT	2004/08/13 15:29
15	10870	707/1-4,10.ccls. or 707/100-102,104.1.ccls. or 707/200,201.ccls.	USPAT	2004/08/17 07:47
16	50	pipelin$4 with travers$3 and partition$2	USPAT	2004/08/17 08:12
17	836	(class adj librar$4) and (object adj oriented)	USPAT	2004/08/17 08:13
18	11	(steady adj state with loop$2) and prologue$2 and epilogue$2	USPAT	2004/08/17 08:33
19	10870	707/1-4,10.ccls. or 707/100-102,104.1.ccls. or 707/200,201.ccls.	USPAT	2004/08/19 13:18

	Hits	Search Text	DBs	Time Stamp ▵
20	1	(linked adj list$2) and sub$2list$2 and (state adj vector$2)	USPAT	2004/08/19 16:08
21	1299	forest with tree$2	USPAT	2004/08/19 16:09
22	8	(forest with tree$2) and (array with pointer$2) and root and node$2	USPAT	2004/08/19 16:14
23	3	(forest with tree$2) and (array with pointer$2) and root and node$2 and travers$4	USPAT	2004/08/19 16:16
24	10831	707/1-4,10.ccls. or 707/100-102,104.1.ccls. or 707/200,201.ccls.	USPAT	2004/08/20 07:59
25	10870	707/1-4,10.ccls. or 707/100-102,104.1.ccls. or 707/200,201.ccls.	USPAT	2004/08/20 07:59
26	14	(class adj librar$4) and (object adj oriented) and (set with macros)	USPAT	2004/08/20 08:18
27	14	(class adj librar$4) and (object adj oriented) and (set with macros)	USPAT	2004/08/20 08:19
28	58	(creat$4 or build$4 or generat$4) with parallel adj2 structure$2 and partition$2	USPAT; US-PGPUB	2004/08/20 08:21
29	1	20040133747.pn.	USPAT; US-PGPUB	2004/08/20 12:49
30	1	queues with swap with roles	USPAT; US-PGPUB	2004/08/20 12:49